agriculture

Review

Advances in Recognition Methods for Fruit and Vegetable Harvesting

Dianlei Han *, Shixing Xu, Siyu Zhou, Qingzhen Zhu and Xuegeng Chen

School of Agricultural Engineering, Jiangsu University, Zhenjiang 212013, China;
2222516070@stmail.ujs.edu.cn (S.X.); 3252502108@stmail.ujs.edu.cn (S.Z.); qingzhen_zhu@ujs.edu.cn (Q.Z.);
1000005204@ujs.edu.cn (X.C.)
* Correspondence: handianlei@ujs.edu.cn

Abstract

The harvesting of fruit and vegetable crops has long been plagued by prominent issues such as high labor costs, low harvesting efficiency, and high fruit damage rates. The application of object recognition technology has enabled harvesting robots to identify, detect, and locate crops in certain agricultural scenarios, achieving a degree of automated harvesting. However, these systems still suffer from shortcomings such as poor robustness in complex environments, insufficient generalization ability, and high model deployment costs, which significantly limit their large-scale application in agricultural harvesting equipment. This paper comprehensively reviews recent literature in the field of fruit and vegetable target recognition. It summarizes how current research focuses on the implementation principles and directions for the improvement of mainstream methods—including digital image processing, traditional machine learning, and deep learning—while also identifying the remaining issues and challenges facing current technology in terms of algorithmic model real-time performance, robustness, and generalization ability. In the future, target recognition technology is expected to achieve breakthroughs through approaches such as multimodal feature fusion, large-scale models, and semi-supervised learning, evolving toward higher accuracy, faster processing speeds, and easier deployment, thereby providing technical support for the large-scale implementation of smart agriculture.

Keywords: agricultural harvesting robot; fruit and vegetable recognition methods; digital image processing; machine learning; deep learning

1. Introduction

According to data from the Food and Agriculture Organization of the United Nations (FAO), the demand for fruit and vegetables is growing rapidly due to global population growth and changes in dietary patterns. They play a significant role in the global food supply, accounting for nearly 30% of total food consumption [1]. To meet this growing demand, countries around the world have been promoting the large-scale and intensive development of agriculture; by 2023, yields per hectare for fruit and vegetable crops had reached 13.5 metric tons and 22.1 metric tons, respectively [2]. However, harvesting—the most labor-intensive stage in fruit and vegetable production—remains predominantly manual in many countries and regions. High labor costs and low operational efficiency make it difficult to meet the demands of large-scale cultivation. At the same time, poor harvesting quality leads to increased losses at various stages of production. These issues have become key bottlenecks constraining the improvement of quality and efficiency in

Academic Editor: Xanthoula Eirini Pantazi

Received: 29 July 2026
Revised: 1 September 2026
Accepted: 1 September 2026
Published: 5 September 2026

fruit and vegetable production, creating an urgent need for breakthroughs in automation technology [3].

To address the challenges of manual harvesting, agricultural harvesting robots have emerged. These robots integrate advanced machine vision, motion control, and flexible harvesting technologies, achieving precise harvesting through the coordination of end-effectors and perception-decision systems [4–9]. Their functions have been expanded to include post-harvest transportation, storage, and grading [10–12], as shown in Figure 1. Within this technological framework, target recognition and detection serve as a prerequisite. They enable robots to accurately distinguish crops from weeds, branches, leaves, and other background elements in complex field environments; acquire 3D pose information; and rapidly assess fruit ripeness to select harvestable targets. In addition to the initial identification of harvestable fruit, the actual physical harvesting process faces a series of daunting engineering challenges. Once identification is complete, the robot must first determine the precise coordinates of the harvesting point and execute complex motion planning to ensure that its end-effector does not cause damage as it navigates through dense branches and foliage; it must gently pick the crop without damaging it, and coordinate with post-harvest processing tasks. These challenges at the physical execution level place stringent demands on the upstream perception system, requiring not only high accuracy but also real-time reasoning capabilities and the ability to be robustly deployed on mobile platforms with limited computational resources. Consequently, all subsequent physical harvesting operations rely on reliable target recognition, which means that recognition performance determines the overall quality and efficiency of the harvesting robot's operations and serves as a core metric for assessing its level of intelligence [13]. Therefore, this paper aims to provide a comprehensive and in-depth analysis of the principles, evolutionary history, and challenges associated with target recognition and detection technologies in harvesting robot operations; subsequent processes at the execution level—such as motion planning—are beyond the scope of this review.

Figure 1. Typical picking robot [4,14–20].

The development of object recognition and detection technology can be broadly divided into three stages. Beginning in the 1980s, significant progress was made in traditional image processing technology. This technology centered on feature extraction based on manually designed rules and achieved object recognition through methods such as threshold segmentation, edge detection, and texture analysis, which were sufficient for recognition applications in simple scenes [21]. The period from the early 21st century to 2010 marked

https://doi.org/10.3390/agriculture16171924

the machine vision phase, during which self-learning classification algorithms—such as Support Vector Machines (SVM) and K-means—were widely used for object recognition and detection. These algorithms enhanced recognition performance by extracting fused features such as color, shape, and texture. After 2010, with continuous breakthroughs in computing power, deep learning technology flourished, and recognition technologies centered on Convolutional Neural Networks (CNN) began to gain prominence. By autonomously learning to extract image features and overcoming the bottlenecks of traditional technologies, deep learning has demonstrated significant advantages. Through deep neural networks that automatically learn the high-level semantic features of target crops, these systems effectively resist disturbances such as changes in lighting and occlusion by branches and leaves, achieving generally high recognition accuracy in complex field environments. Furthermore, building on recognition capabilities, they can simultaneously perform other tasks such as variety identification, maturity grading, disease detection, and field management [22–29]. The emergence of these algorithms has propelled target recognition technology for harvesting from experimental research to practical implementation.

Given the importance of target recognition technology in harvesting robots and the rapid pace of current technological development, it is necessary to comprehensively review and summarize research findings from recent years. Although several relevant reviews have been published [30–37], the following limitations are present: first, some studies cover only traditional image processing and machine learning methods, failing to systematically incorporate the rapid advancements in the field of deep learning in recent years; second, most focus solely on a single crop, a single type of algorithm, or a single visual task, lacking a comprehensive comparative analysis of the entire technological trajectory from traditional methods to deep learning; most critically, existing studies predominantly adopt a technical perspective, evaluating algorithm performance solely based on general detection metrics such as accuracy and Mean Average Precision (mAP), without conducting an in-depth analysis of how these metrics impact harvesting outcomes. For instance, inference speed directly determines the operational efficiency and yield per unit time of harvesting robots; the degree of model lightweighting affects the hardware costs and battery life of mobile harvesting platforms; and localization and segmentation accuracy directly influence the picking success rate of end-effectors and the rate of mechanical damage to fruit. To address these shortcomings, this paper focuses on fruit and vegetable harvesting scenarios and, based on recent domestic and international literature, provides a comprehensive review that explains the principles, evolution, and applicable scenarios of three mainstream technologies: digital image processing, traditional machine learning, and deep learning; Focusing on the practical needs of harvesting, this paper summarizes the directions for improvement and application effectiveness of mainstream algorithms in addressing core harvesting challenges—such as data annotation costs, edge deployment constraints, and field environment interference—and conducts an in-depth analysis of existing challenges from three dimensions: algorithm performance, deployment costs, and environmental adaptability. It also outlines future development trends by integrating cutting-edge technologies such as multimodal feature fusion and semi-supervised learning. Compared to existing reviews, this paper consistently uses the practical needs of harvesting operations as the evaluation benchmark, interpreting the application value and limitations of various recognition technologies from the perspective of harvesting performance. It aims to clarify the current state of development in fruit and vegetable harvesting recognition technology and provide a systematic theoretical reference for subsequent research and engineering implementation.

https://doi.org/10.3390/agriculture16171924

2. Methodology

To ensure the systematic and rigorous nature of this review, this paper comprehensively organizes and categorizes the research literature related to object recognition in the field of fruit and vegetable harvesting robots and establishes detailed strategies for literature retrieval, screening, and evaluation. The overall process follows the methodology for systematic literature reviews and primarily consists of three stages: literature search, inclusion and exclusion criteria, and systematic evaluation, as shown in Figure 2.

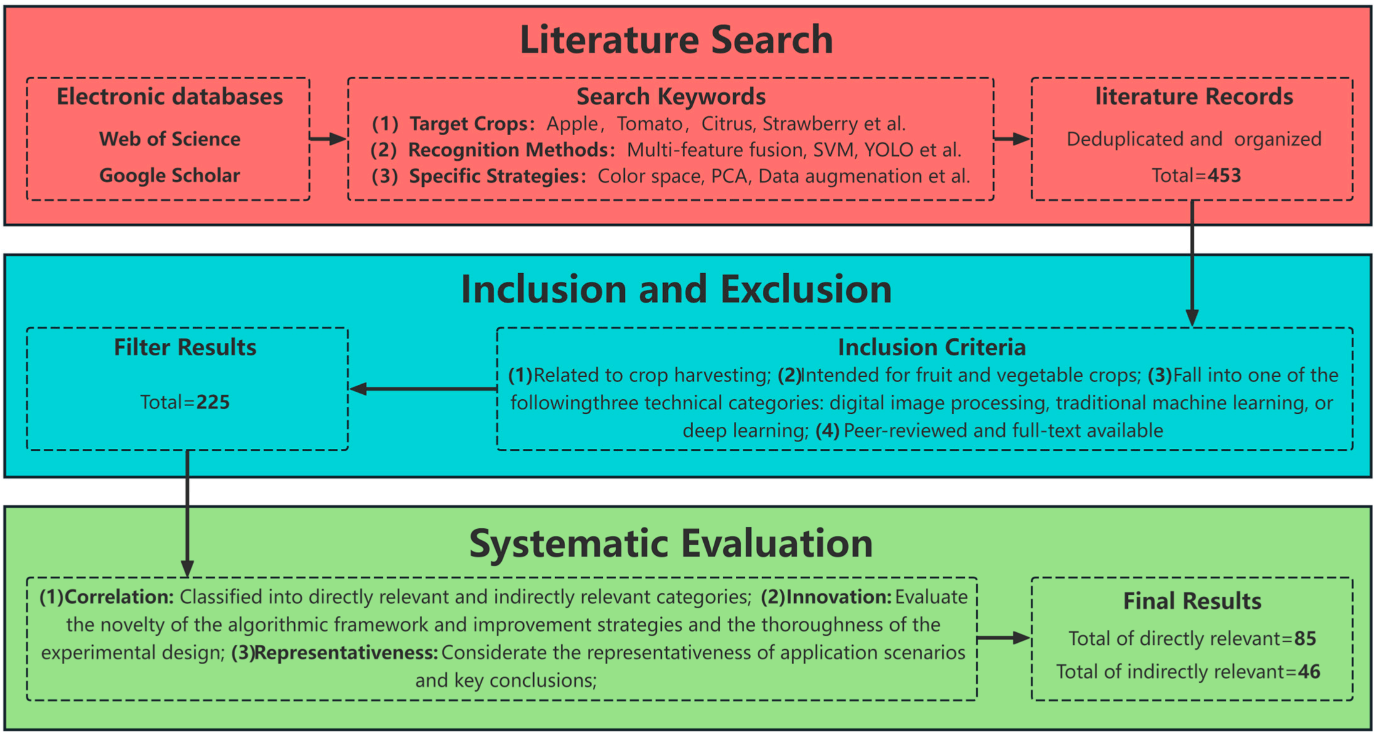


Figure 2. Overview of literature review methods.

The literature search phase aims to identify as many candidate publications related to the research topic as possible. This study primarily uses Web of Science and Google Scholar as search databases. Web of Science includes high-impact journals spanning fields such as agricultural engineering, computer science, and robotics; its comprehensive bibliographic information facilitates subsequent citation tracking and management. Google Scholar, on the other hand, offers broader coverage of academic resources, enabling the retrieval of other outputs, including technical reports and non-English literature. The publication period for the retrieved literature was primarily set from 2017 to 2026, covering the latest application results of target recognition algorithms in harvesting in recent years. Search keywords encompassed three aspects: target crops, recognition methods, and specific strategies. To ensure the academic quality of the literature, the scope of journals was primarily limited to Q1/Q2 journals in various disciplinary fields and papers from authoritative conferences in agricultural engineering and robotics; during subsequent screening, priority was given to research with comprehensive experimental designs and standardized evaluation metrics. Additionally, the search did not impose excessive restrictions on countries or regions to include research findings from diverse growing environments and climatic conditions. For literature that was thematically relevant but deviated from the primary focus during the search, specific keywords were excluded to narrow the scope to harvesting scenarios. After completing the database search, all literature records were deduplicated and preliminarily organized.

The inclusion and exclusion criteria were designed to select studies from the retrieved candidate literature that were highly relevant to the theme of the review, employed standardized research methods, and provided comprehensive information. To this end, this paper established clear inclusion criteria: (1) the research subject is fruit or vegetable crops; (2) the research content involves visual recognition technologies in the harvesting process, including but not limited to object detection and segmentation, ripeness assessment, 3D localization, and picking point detection; (3) the research methods fall into one of the following three technical categories: digital image processing, traditional machine learning, or deep learning; (4) the study was published in a peer-reviewed journal or conference as a full-text article.

After completing the inclusion and exclusion screening, this paper further conducted a systematic evaluation of the ultimately included literature and categorized the relevant studies into two types: directly related and indirectly related. Directly related studies focus on core tasks such as identification, localization, and ripeness assessment during harvesting operations, and concentrate on algorithmic improvements and performance enhancements under various environmental conditions; these studies constitute the primary subjects of analysis in this review. Indirectly relevant studies, on the other hand, include tasks such as variety classification, disease detection, and field management—which influence harvesting decisions but do not directly involve crop recognition. While these studies are not the primary focus of analysis, they are included as reference material when discussing the extended capabilities of recognition technologies or comparing the applicability of different methods. During this phase, each paper was reviewed in detail with regard to its abstract, the algorithmic framework employed, specific improvement strategies, experimental design, application scenarios, and main conclusions. The value of inclusion was comprehensively assessed based on three dimensions—relevance, innovation, and representativeness—and papers with outstanding contributions were discussed in detail in subsequent chapters. Following the screening and evaluation across the three phases described above, a total of 131 papers were ultimately included in the analysis, comprising 85 directly relevant studies on harvesting target recognition and 46 indirectly relevant reference studies.

3. Recognition Methods Based on Digital Image Processing

Digital image processing technology relies on the differences in visual features between objects and their backgrounds. It achieves object segmentation and recognition by using artificially designed operators to extract key features such as color, shape, and texture. From a recognition mechanism perspective, many mature fruit and vegetable crops exhibit significant differences in various features compared to the surrounding branches, leaves, and soil, and this high level of distinguishability provides the foundation for the recognition and detection of these crops. To provide a systematic overview of these feature-based approaches, Table 1 summarizes the mainstream feature categories—color, shape, and texture—along with their corresponding extraction methods, applicable crops, and typical advantages and limitations. The following three subsections elaborate on each category in detail, examining their implementation principles, application scenarios, and performance boundaries in harvesting operations.

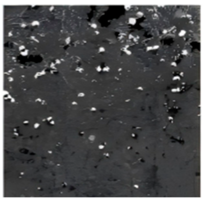
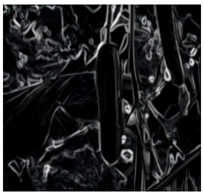
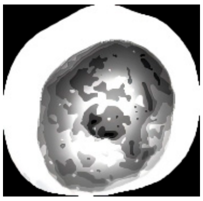
3.1. Methods Based on Color Features

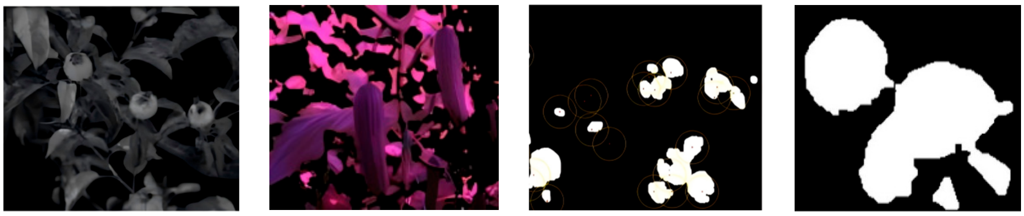
In terms of color features, mature fruits and vegetables—such as apples, strawberries, and tomatoes—stand out easily against the green of foliage and the gray-brown of soil. The three channels of the raw RGB color space combine color and brightness information, making them relatively sensitive to exposure, shadows, and uneven lighting in natural

https://doi.org/10.3390/agriculture16171924

environments. Existing research often begins by optimizing the structure or channels of the RGB space, or converting it to other specific color spaces to obtain independent components, and then uses methods such as histograms, color moments, and threshold segmentation to convert color information into quantifiable parameters. Figure 3 shows segmentation images under different color spaces and components.

Table 1. Comparative analysis of recognition methods based on digital image processing [38–40].

Feature Category	Feature Extraction Methods	Applicable Crops	Advantages	Disadvantages
Color-based methods	 Color space conversion, single-channel component extraction, color moments, color histogram	Apple, citrus, strawberry, tomato, grape	Good performance for crops with distinct color differences; low computational cost and fast detection speed	Not suitable for scenes where the crop and background are similar in color; it is relatively sensitive to changes in exposure, shadows, and lighting
Shape-based methods	 Sobel/Canny edge detection, contour descriptors, Hough transform, geometric moments	Cucumber, pepper, eggplant, mango	Less affected by illumination; strong geometric discriminability	Unsuitable for occlusion/overlap scenarios; highly sensitive to crop growth stochasticity
Texture-based methods	 Gray-level co-occurrence matrix (GLCM), Local Binary Pattern (LBP), Gabor transform	Apple, citrus, lychee, pineapple, kiwifruit	Effectively separates targets and backgrounds that are similar in color or overlapped/occluded	High computation, slow speed, illumination sensitivity



(**a**) (**b**) (**c**) (**d**)

Figure 3. Examples of color segmentation for selected fruit and vegetable crops [38,39,41,42]: (**a**) Green apple image segmented by adaptive G-B color difference; (**b**) Cucumber segmentation result in HSV color space; (**c**) Ripe apple segmentation using the a-channel of Lab color space; (**d**) Bayberry segmentation using Otsu's method with R-G color difference.

Biffi et al. systematically compared the segmentation performance of four color spaces—normalized RGB, HSV, CIE L*a*b*, and improved OHTA—for mature apples in natural scenes. Among these, the CIE L*a*b* space achieved the best detection accuracy of 76% in the a-channel; however, the algorithm was unable to segment clusters of adhered fruits into individual fruits, and the reddish-brown soil exposed between rows also led to missed detections [38]. Xu et al. converted the RGB color space to the HSV color space to rapidly perform a coarse screening of strawberry regions, achieving an accuracy rate exceeding 98% and filtering out the vast majority of background interference in advance [43]. To address the issues of uneven canopy illumination and the similarity in color between green and yellow bayberries and leaves, Lei et al. used the YUV color space to extract only the Y component for block-based Contrast Limited Adaptive Histogram Equalization (CLAHE). They also proposed an adaptive color difference formula for the coarse segmentation of bayberries at various stages of ripeness, achieving an accuracy of 97.4% in tests using natural orchard images [41]. Li et al. selected the Cb channel in the YCbCr color space—which exhibits the most significant grayscale difference between the target and the background—for processing. They performed threshold segmentation by fusing pixel grayscale values with neighborhood spatial information, thereby resolving the issues of threshold shift in traditional Otsu segmentation caused by uneven illumination and false positives resulting from the similarity in color between branches, leaves, and lychees [44]. Xia et al. constructed a green-blue difference formula using the global mean of the G and B channels in a single image as the adaptive coefficient, significantly increasing the grayscale contrast between green apples and the background branches and leaves. They combined this with an iterative threshold segmentation algorithm to perform binary segmentation on the adaptive color difference map; this method achieved an accuracy of 93.13% on a test set of 108 green apple images [42].

Methods based on color features essentially cluster similar pixels within a specific color space, but the color of an object captured by a sensor is not stable; it is influenced by factors such as lighting and the reflective properties of the object's surface. In RGB images, luminance and chrominance information are coupled; uneven lighting and localized surface shadows can cause significant changes in color vectors, causing the target's color features to converge with those of the background. Even when converting to a color space where luminance and chrominance are decoupled to isolate components that are more robust to luminance fluctuations, or when using local enhancement algorithms to compensate for underexposed shadow areas, severe lighting or shadow interference can still lead to instability in the calculation of chrominance components. Occlusion and overlap not only result in incomplete object contours and fragmented edge color gradients but also reduce the effective light exposure of inner fruit layers, creating local shadows that shift their chromaticity toward the background and cause missed detections in threshold segmentation. Color similarity interference falls into two categories: first, partial spectral overlap in a single channel, where the target and background exhibit similar responses in a specific color channel but remain distinguishable in other chromatic dimensions (this can be resolved by selecting an optimal color channel); second, synchronous spectral overlap, where the RGB responses of the target and background drift in sync under changing illumination, rendering fixed color difference formulas ineffective and necessitating the introduction of adaptive channel weighting strategies. In summary, methods based solely on color features remain practical in application scenarios—such as greenhouses—where lighting conditions are relatively stable and the color difference between the target and the background is significant, due to their intuitive principles and computational efficiency. However, light conditions in natural field environments fluctuate dramatically, and severe obstruction and shadowing can still cause the algorithm to produce false positives and false negatives.

3.2. Methods Based on Shape Features

Most fruit and vegetable crops exhibit regular geometric shapes—such as circular, elliptical, or elongated forms—that contrast with the irregular contours of branches and leaves, providing an additional criterion for identification and detection. The implementation logic is similar to that for color features: edge detection operators such as Sobel and Canny are used to extract the object's contour and output a discrete set of edge points, which are then converted into quantifiable parameters using methods such as contour descriptors, the Hough transform, and geometric moments.

Within the pre-screened candidate regions, Liu et al. used the Sobel operator and Fourier descriptors to fully preserve the global shape features of cucumbers. They performed gradient vector similarity matching against a pre-generated cucumber contour template; at the same time, normalizing the gradient vectors effectively suppressed the impact of lighting on similarity calculations. The overall recognition accuracy reached 86%, with a real-time frame rate of 38.9 FPS, meeting the operational requirements of cucumber harvesting robots [39]. Zhang et al. proposed a region-marked gradient Hough transform. Compared to the traditional Hough transform, this method performs circle detection only within the minimum bounding rectangle of the apple's contour, thereby avoiding a full-image scan and effectively mitigating the attenuation of edge gradients caused by lighting conditions. As a result, recognition accuracy improved by 15.03% and 16.41% under front-lit and back-lit conditions, respectively [45]. Trang et al. utilized the Elliptical Fourier Descriptor (EFDS) to quantify the irregular contour of the tomato's base into numerical features, fully preserving geometric information such as contour curvature, and achieved the recognition and classification of different tomato varieties; however, occlusion can significantly alter the extracted object boundaries [46]. Yin et al. proposed an improved Fuzzy C-Means (FCM) segmentation method and an enhanced circular Hough transform algorithm, introducing similarity metrics based on both pixel grayscale relationships and spatial relationships. By fitting the set of boundary points of the apple's pixel region, they achieved accurate apple recognition in complex scenes [47]. Barnea et al. combined point cloud edge extraction and specular highlight detection algorithms, using an RGB-D depth camera to extract 3D shape features and 2D specular highlight features to identify and localize bell peppers. Compared to full-space 3D scanning, this approach increased speed by 43%; however, the camera generates depth noise under strong lighting, and the average accuracy drops to only 55% when there are numerous occluded objects [48].

Although shape-based approaches reduce sensitivity to illumination-induced color shift, their performance hinges heavily on intact target contours. Strong backlight and shadows do not alter physical geometry but corrupt edge gradient responses and generate spurious edges. Occlusion and overlap are the primary causes of algorithm failure: for edge-fitting algorithms, occlusion results in the loss of a large number of valid edge pixels or surface points in the point cloud, and edge fragmentation leads to fitting errors and failure; for contour descriptors, non-closed or incomplete contours directly distort curvature-related shape coefficients, reducing classification accuracy. Compared to color-based features, shape-based recognition methods are less affected by changes in lighting and offer stronger geometric discriminative power. However, in real-world harvesting scenarios, targets often lack complete and clear contours, and occlusions and overlaps increase the difficulty of shape feature extraction; therefore, these methods must be combined with other features.

3.3. Methods Based on Texture Features

The smoothness or specific texture of fruit surfaces differs from the vein patterns on leaves and branches in terms of grayscale distribution patterns, which provides the basis for object segmentation. Such methods extract the textural features of objects using techniques such as GLCM, LBP, and Gabor Transforms, and identify objects based on the grayscale distribution patterns in local regions; they can still achieve effective matching even when objects are similar in color or shape, or when partially occluded. Figure 4 illustrates the principles behind two widely used texture feature extraction methods.

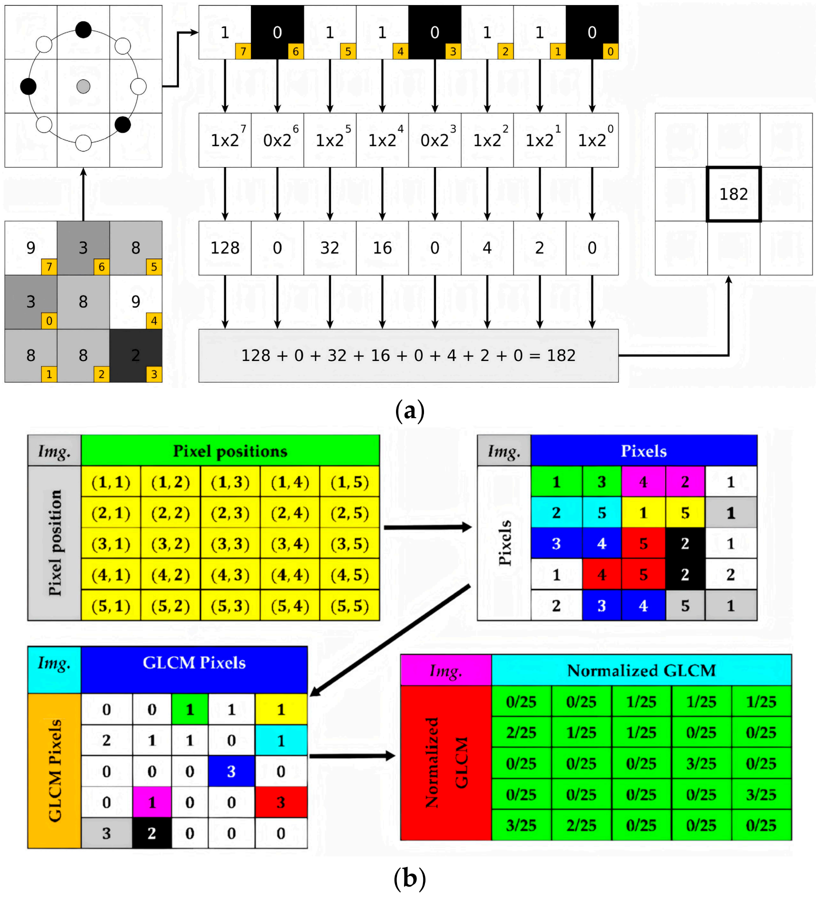


Figure 4. Typical texture feature extraction approaches [49,50]: (**a**) The Calculation Process for LBP Descriptors; (**b**) The Calculation Process for GLCM.

To address the classification of citrus fruit from different origins that are similar in appearance, Nanda et al. used the GLCM to extract six texture features from laser backscatter images: contrast, dissimilarity, homogeneity, energy, angular second-order moment, and correlation. They calculated the eigenvalues in four directions and took their average, achieving a classification accuracy of 96.667% [51]. Ropelewska collected scanned images of various apple surfaces—including the skin, longitudinal sections, and cross-sections—extracted texture features from multiple color channels, and combined these with a multilayer perceptron and a multi-class classifier to achieve rapid apple variety identification using only a flatbed scanner [52].

Texture features can also be used for the phenotypic detection of targets. Yang et al. used Adaptive Bilateral Filtering and Gamma Correction to enhance texture distinctiveness. They then employed the GLCM to extract five texture features—angular second-order moment, entropy, contrast, inverse difference moment, and correlation—for apple surface defect detection. This algorithm achieved a recognition accuracy of 93.7% for fruit stems and calyxes, and 94.2% accuracy for surface scar detection [40]. Shen et al. employed LBP to extract texture features and constructed a detection model for pepper blight using only the mean values of LBP features from 10 feature wavelengths with low noise; the accuracy rates on the training and test sets reached 98.96% and 100%, respectively, outperforming recognition and detection models based on spectral features [53].

Although texture-based methods have proven effective for fruit phenotype detection, they are computationally complex and sensitive to changes in lighting conditions, posing challenges for practical deployment. Compared to methods based on color or shape features, texture descriptors extract features from multiple channel dimensions; the sharp increase in feature vectors and repetitive computational processes directly leads to a significant increase in training and inference times. Furthermore, both GLCM and LBP rely directly on absolute or relative grayscale values. When lighting conditions change, the grayscale values or the direction of grayscale gradients on the same physical surface can change significantly, and the non-uniform surface reflectance of an object may introduce spurious grayscale patterns that are difficult to distinguish from true texture features. The aforementioned studies either conducted experiments under strictly controlled lighting conditions or invested substantial algorithmic resources in lighting pre-correction, which increased the overall computational cost without completely eliminating the risk of false positives under complex lighting conditions. Therefore, while texture-based recognition methods can help harvesting robots remove diseased fruit and sort crops during operation, their applicability in unstructured agricultural scenarios still requires further evaluation.

3.4. Methods Based on Multi-Feature Fusion

Due to the complexity of agricultural scenarios, object recognition based on a single feature often has significant limitations. Relying solely on color features is susceptible to changes in lighting and interference from shadows and is not suitable for scenarios where the object and background are similar in color. Relying solely on shape features makes it difficult to effectively characterize the specific form of an object when it is partially occluded, overlapped, or has an incomplete outline. Relying solely on texture features, on the other hand, is prone to misclassification due to image resolution and noise sensitivity. Therefore, most current object recognition methods based on image processing technologies employ multi-feature fusion.

The fusion of color and shape features is the earliest and most widely adopted approach. Ifmalinda et al. used the r-component of the normalized RGB color space to classify the ripeness and color of oranges; they segmented the fruit region through global threshold binarization and established a linear regression model to convert pixel area into actual physical diameter, thereby achieving fruit size classification [54]. Zhang et al. compared and validated the segmentation performance of three color spaces—RGB, HSV, and Lab—and ultimately selected the a* component of the Lab space, which offers the best discrimination between red and green hues, combined with the circular Hough transform to detect citrus fruit. Within the optimal shooting distance of 1.5–2.1 m, the detection accuracy exceeded 90% [55]. Yang et al. utilized 2R-G-B color features and Otsu thresholding and employed a locally parameterized adaptive Hough circle transform to rapidly identify apples in the initial frame, achieving a recognition success rate of 91.3% under laboratory conditions with an average processing time of 72 ms [56]. In terms of fusing color and texture features, Yu et al. used color features and depth information to quickly filter out distant backgrounds and invalid regions and combined these with improved LBP texture features to distinguish green lychees from leaves; this method supports the detection of both ripe and unripe lychees [57]. Kurtulmus et al. first removed most of the background based on color histograms, then used the "fruit feature" method—which analyzes brightness, saturation components, and circular Gabor textures—to determine whether a subwindow contained a fruit, achieving a detection rate of 75.3% on a validation set of unripe green citrus fruit [58]. Regarding the fusion of shape and texture features, Lu et al. first used multiscale LBP texture features to filter out most background noise to distinguish unripe citrus fruit from leaves. They then employed Hierarchical Contour Analysis (HCA) to analyze the brightness distribution of multiple concentric circles centered on highlights, followed by a circular Hough transform fit, achieving an accuracy rate of 82.3% on their test set [59].

The aforementioned studies on feature fusion based on digital image processing exhibit a high degree of consistency in their implementation methods, all following a logic that progresses from coarse screening to fine discrimination: first, a single feature is used to quickly filter out large areas of background noise, extract candidate regions, and narrow the detection scope; then, the remaining features are used to perform fine-grained verification of the candidate regions and eliminate false targets (Table 2).

Table 2. Comparison of various feature fusion strategies.

Fusion Method	Primary Features	Secondary Features	Advantages	Disadvantages
Color-Shape	Color	Shape	Low computational complexity; good real-time performance	Cannot handle extreme scenarios where targets and background share the same color and shape; under severe occlusion or adhesion of fruits, the contour may break and shape verification fails.
Color-Texture	Color	Texture	Combines the efficiency advantage of color segmentation with the detail discrimination capability of texture	No significant improvement for occlusion and adhesion problems of fruits
Shape-Texture	Shape/Texture	Shape/Texture	Suitable for scenes where color is similar to background; robust to changes in light conditions	High computational complexity, poor real-time performance; severe occlusion leads to failure of both detection stages; difficult parameter tuning
Color-Shape-Texture	Color/Shape	Texture	Can resist multiple types of interference simultaneously; offers the best overall recognition accuracy and robustness	Large computational load, poor real-time performance; high feature redundancy, complex parameter tuning;

The emergence of this architecture is no coincidence; it stems from the fact that harvesting robots are typically equipped with embedded processors of limited computational capacity and must complete image processing and decision-making within a very short time frame to ensure operational efficiency. Since color features involve only independent operations on a per-pixel basis, they have the lowest computational complexity and the highest global segmentation efficiency. Furthermore, the color difference between ripe fruit and the background is typically significant; therefore, most fusion schemes use color features as a coarse-screening tool in the first stage. However, the discriminative information provided by color features is relatively limited, so texture or shape features must be introduced in the second stage to compensate and ensure the algorithm's effectiveness. Since the computational complexities of shape- and texture-based features differ only slightly, their order of application depends on the type of interference present in the scene. This structure enhances the robustness and real-time performance of feature fusion algorithms, meeting the requirements for the efficient operation of harvesting robots in most simple scenarios. However, when faced with multiple crops or strong interference, the lack of feature expressiveness and generalization ability limits the robot's operational capabilities in complex natural environments such as fields and orchards.

4. Recognition Methods Based on Traditional Machine Learning

Traditional machine learning-based object recognition classifies objects through autonomous learning based on manually designed features. By integrating multidimensional features and model training, it enhances generalization capabilities; its development and maturation have shifted object recognition methods from rule-driven to data-driven approaches. Compared to digital image processing technologies, machine learning algorithms offer significantly improved accuracy and robustness in complex scenarios such as those with complex backgrounds and fluctuating lighting conditions. However, their recognition performance remains dependent on manually designed features, and the selection, fusion, and filtering strategies for feature dimensions directly determine the upper limit of a model's performance. Furthermore, when handling complex tasks, the increase in feature dimensions and sample size leads to higher training and computational complexity, making parameter selection and optimization more challenging. Currently, mainstream classifiers based on traditional machine learning are only applicable to a single crop or specific task scenarios and lack sufficient generalization ability. To address these issues, many researchers have conducted extensive and in-depth studies using various classifiers such as SVMs, K-means, and Random Forest (RF).

4.1. Methods Based on SVM Classifier

An SVM achieves separation of the target from the background by finding the optimal classification hyperplane in a high-dimensional feature space. It performs exceptionally well in scenarios with few samples and high-dimensional features and is a widely used supervised learning algorithm in the field of crop recognition.

Fujinaga et al. innovatively proposed a method for generating a tomato growth state atlas and investigated the optical properties of tomatoes under near-infrared imaging. After extracting Histogram of Oriented Gradients (HOG) features, they used an SVM classifier to perform fruit recognition; experiments showed that the F1 score for fruit recognition reached 0.847 [60]. Bai et al. proposed an object recognition algorithm based on the morphology and growth characteristics of clustered tomatoes. By extracting and fusing HOG shape features, LBP texture features, and color features from multiple color spaces, they fed the data into an SVM classifier to achieve precise fruit recognition. This recognition model achieved 100% precision, recall, and accuracy, with each image recognized in less than 1 s [61]. Lin et al. proposed a general-purpose fruit detection framework based on partial shape matching and the probabilistic Hough transform. By training an SVM classifier using Color-HOG features to filter out false positives, this method achieved excellent detection accuracy and recall rates on a dataset comprising six types of fruit—citrus, tomatoes, pumpkins, bitter melons, loofahs, and mangoes—which vary in color and shape [62]. Baculo et al. developed an HOG-SVM mango detection model, achieving unified identification of healthy and defective mangoes for the first time. This study extracted HOG features only from color-segmented candidate regions and achieved an F-score of 89.38% using linear SVM classification, with an average IoU of 0.7938 for candidate segmentation [63].

To address the issue where the complex distribution of texture features in green citrus and leaves prevented the original SVM from distinguishing between them, Zhao et al. proposed a green citrus detection framework combining Adaptive Red-Blue Difference (ARB) maps with the Sum of Absolute Transformed Difference (SATD) block matching. They trained a Gaussian radial basis kernel SVM classifier using five selected texture features to filter out false-positive regions, achieving a recognition accuracy of 83.4% on the test set [64]. Peng et al. developed a multi-class SVM model based on the Error-Corrected Output Code (ECOC) framework capable of classifying five grape varieties, while addressing the issues of classification instability in small-sample scenarios and performance degradation caused by high-dimensional features [65].

In summary, as the most widely used and mature traditional machine learning classifier, the SVM has two main advantages: first, it delivers stable classification performance in low-sample-size scenarios without being prone to overfitting, and possesses good generalization ability; second, it can effectively handle complex nonlinear problems and high-dimensional data. The core principle of an SVM is to maximize the margin between classes; the model's complexity is determined solely by a small number of support vectors rather than the size of the dataset, enabling it to effectively resist noise and outliers. Additionally, the introduction of a kernel function maps the data to a higher-dimensional feature space, making it linearly separable, while computing the inner product directly in the low-dimensional space does not increase computational complexity. However, the training complexity of SVMs increases significantly as the number of samples grows. Their classification performance is relatively sensitive to the choice of kernel function and parameter optimization, and feature redundancy and noise directly affect the accuracy of the classification boundary. Furthermore, since SVMs are fundamentally designed to solve binary classification problems, an increase in the number of classes will significantly reduce model performance and recognition accuracy.

4.2. Methods Based on K-Means Clustering

As a typical unsupervised learning algorithm, the K-means clustering algorithm achieves object segmentation by automatically clustering similar pixel features, eliminating the need for manually labeled training data and offering unique advantages in data-scarce scenarios. However, traditional clustering suffers from drawbacks such as sensitivity to initial cluster centers and a tendency to get stuck in local optima. To enable real-time eggplant harvesting decisions on low-computational-power edge devices, Tamilarasi et al. developed a detection algorithm combining HSV background removal, K-means pixel clustering, and symmetry testing, achieving a detection accuracy of 87.48% on a Raspberry Pi device [66]. Wang et al. designed an adaptive initial cluster center selection algorithm based on neighborhood hue similarity and constructed a clustering feature space that weight-fused multiple features, including color, texture, and morphology. Experimental results show that this method achieved a recognition accuracy of 93%, maintained a detection accuracy of over 91% even when fruit overlap exceeded 40%, had a 3D localization error of less than 1%, and demonstrated superior real-time performance and robustness compared to deep learning models such as You Only Look Once version 7 (YOLOv7) and Mask Region-based Convolutional Neural Network (Mask R-CNN) [67]. Wakchaure et al. used the K-means algorithm for image segmentation, separating the sapodilla fruit regions from the background in the original image through pixel-level color clustering to generate precise regions of interest. The coefficient of variation for the R and G channel color features extracted after segmentation was only 2.77–2.86% [68].

To address the challenges of distinguishing sweet peppers from their foliage background in greenhouse environments and the high cost of obtaining annotated data, Doosti-Irani et al. fused 3D geometric features derived from fast point feature histograms with HSV color features. After subsampling, outlier removal, and dimensionality reduction via Principal Component Analysis (PCA), they employed K-means clustering to achieve sweet pepper object recognition, achieving a detection accuracy of 95.10% as evaluated manually [69].

In summary, the K-means algorithm does not require manual labeling; it performs unsupervised clustering based solely on the data's own distance metrics, and the computational complexity of a single iteration is nearly linear. When data clusters are dense and clearly separated from one another, it can achieve ideal clustering results. However, its shortcomings are equally prominent: first, the algorithm requires the parameter K to be specified in advance, and the value of K cannot be automatically inferred from the data; second, the clustering results are highly sensitive to the initial cluster centers, and different initial points tend to converge to different local optima; third, since cluster centers are calculated as averages, outliers or noise can significantly skew their positions, thereby affecting the clustering results. These characteristics not only enable K-means to achieve good clustering results under specific data patterns but also limit its application in scenarios with complex distributions.

4.3. Comparative Analysis of Multiple Traditional Machine Learning Algorithms

For different crop varieties, environmental conditions, and task objectives, many researchers have conducted comparative analyses of the recognition accuracy, scene robustness, and computational efficiency of various classical classification algorithms to identify the classification models best suited for specific tasks. Table 3 summarizes ten comparative studies of traditional machine learning algorithms for agricultural image recognition and detection tasks. The detection targets cover a variety of fruits and vegetables, such as pineapples, apples, and broccoli, while the algorithms examined include several mainstream methods, such as SVMs, K-Nearest Neighbors (KNN), and RFs. These comparative studies, conducted in typical agricultural scenarios, quantify the performance boundaries of different classifiers, providing direct guidance for algorithm selection in similar research and reducing the costs of trial and error. At the same time, they explore the patterns of feature-classifier matching, offering experimental support for subsequent feature engineering optimization and algorithm improvements.

Therefore, although Table 3 summarizes the performance of various classifiers across different agricultural detection tasks, direct comparisons of accuracy values across studies should be interpreted with caution. Significant differences in data collection methods, imaging conditions, sample sizes, evaluation metrics, and experimental designs render cross-study comparisons statistically invalid. Instead, the value of these comparative studies lies in observing the relative performance of different classifiers within each study and understanding the underlying reasons for their superior performance under specific conditions. From this perspective, an SVM, with its principle of minimizing structural risk, demonstrates robust performance in scenarios with small sample sizes and low feature dimensions; whereas RF, through its bootstrap sampling and random subspace mechanisms, exhibits excellent resistance to overfitting in pixel-level classification and multi-feature fusion tasks. Furthermore, the performance of computationally intensive models such as Artificial Neural Network (ANN) and Adaptive Boosting (AdaBoost) is highly dependent on data volume and tuning strategies, making them less suitable for resource-constrained real-time harvesting robots. More fundamentally, the upper limit of any classifier's performance is determined by the data modality and imaging conditions; algorithms that achieve near-perfect accuracy in controlled laboratory environments often experience significant performance degradation in natural environments. Some studies have also shown that a judicious combination of multiple classifiers can fully leverage their respective strengths, thereby yielding better classification results than a single model. At the same time, compared to detection methods based on a single feature, integrating multidimensional features typically improves the accuracy of all classifiers. Image preprocessing techniques can eliminate redundant noise and further optimize the recognition performance of various models. This indicates that the performance of traditional machine learning algorithms still depends on the quality of manually designed features, and their robustness and generalization capabilities in complex situations still need improvement.

Table 3. Studies with comparative analysis of multiple machine learning algorithms.

Research Objective	Data Collection	Extracted Features	Compared Classifiers	Performance	Ref.
Identification and counting of pineapple crowns in plantations	1300 RGB frames captured by a UAV from an aerial perspective under natural lighting conditions	RGB color features, 8 shape features, LBP and GLCM texture features	ANN, SVM, RF, NBDT [1], KNN	Accuracy 94.4%	[70]
Identification of peach varieties	480 RGB images taken with a DSLR camera under artificial lighting against a white background	HSV color features, 14 shape features	SVM, RF, MLP [2]	Classification accuracy 98.7%; average F1 score 0.98	[71]
Apple image segmentation in orchards	105 RGB images captured by a camera under artificial lighting at night	LAB-B, HSV-S, YUV-Y color features, GLCM texture features	NN [3], SVM, DT [4], RF, Adaboost, NB [5], QDA [6]	Test set: accuracy 94%, true positive rate 90%; Orchard images: segmentation error 7%, false positive rate 13%, false negative rate 15%	[72]
Identifying ripe fuji apples in the orchard	52 sets of point cloud data captured by a depth camera under natural lighting conditions	Point cloud, RGB, HSI color features, 3D geometric features	GA-SVM, SVM, KNN, RF	Accuracy 92.30%, recall 89.83%, precision 94.64%; classification accuracy for occluded apples 73.46%	[73]
Field detection, localization, and size estimation of broccoli	214 3D images extracted from videos captured by a TOF camera under natural lighting and LED fill lighting	Point cloud, 3D geometric features	KNN, SVM	UK dataset: average precision 95.2%; Spain dataset: average precision 84.5%	[74]
Detection of immature green bananas	700 RGB images captured by a digital camera under various natural lighting conditions	LBP texture features	SVM, Adaboost	Average detection time 1.325 s; detection rates: 85.71% (front lighting), 89.77% (back lighting), 92.31%(cloudy),	[75]
Non-destructive identification of mango varieties	894 spectral images acquired by a spectrometer under controlled indoor conditions	Infrared reflectance spectral features	SVM, LDA, RF, NN	Variety recognition accuracy 97.44%	[76]

[1] Neural-Backed Decision Trees. [2] Multi-Layer Perceptron. [3] Neural Network. [4] Decision Tree. [5] Naive Bayes. [6] Quadratic Discriminant Analysis.

5. Recognition Methods Based on Deep Learning

Unlike traditional machine learning methods, deep learning algorithms can autonomously learn abstract representations ranging from low-level pixel features to high-level semantic features without the need for manually designed feature rules. Based on a multi-layer neural network framework, they form an end-to-end, hierarchical feature extraction mechanism and directly establish a mapping relationship between image inputs and detection results, simplifying the cumbersome process of feature extraction followed by classification found in traditional machine learning. This fundamental paradigm shift enables them to demonstrate significantly superior generalization capabilities and robustness when handling complex, unstructured scenarios in agricultural applications. With the introduction of AlexNet, CNNs, which possess powerful local feature extraction capabilities, quickly became the mainstream architecture for crop object recognition. As technology advanced, the emergence of DEtection TRansformer (DETR) in 2020 introduced the Transformer architecture into the field of object detection. By utilizing self-attention mechanisms to capture the overall context of an image, it offers a novel approach to agricultural vision tasks. Various mainstream detection algorithms derived from these two architectural categories have been widely applied to agricultural vision tasks.

5.1. Methods Based on the CNN Architecture

Deep learning algorithms based on CNN architectures extract local features by sliding convolution kernels across images. Through hierarchical convolution and pooling operations, they progressively abstract higher-level representations—from edges and textures to semantic objects—enabling end-to-end object detection. In object detection tasks, CNN-based algorithms can be categorized into two-stage and single-stage detection algorithms based on their detection workflow, as illustrated in Figure 5. Two-stage algorithms first generate candidate regions, then perform classification and location regression on those regions; representative methods include R-CNN, Fast R-CNN, and Faster R-CNN. Single-stage algorithms, on the other hand, omit the separate candidate region generation stage and directly output class and location information through the network; representative methods include the YOLO series and SSD.

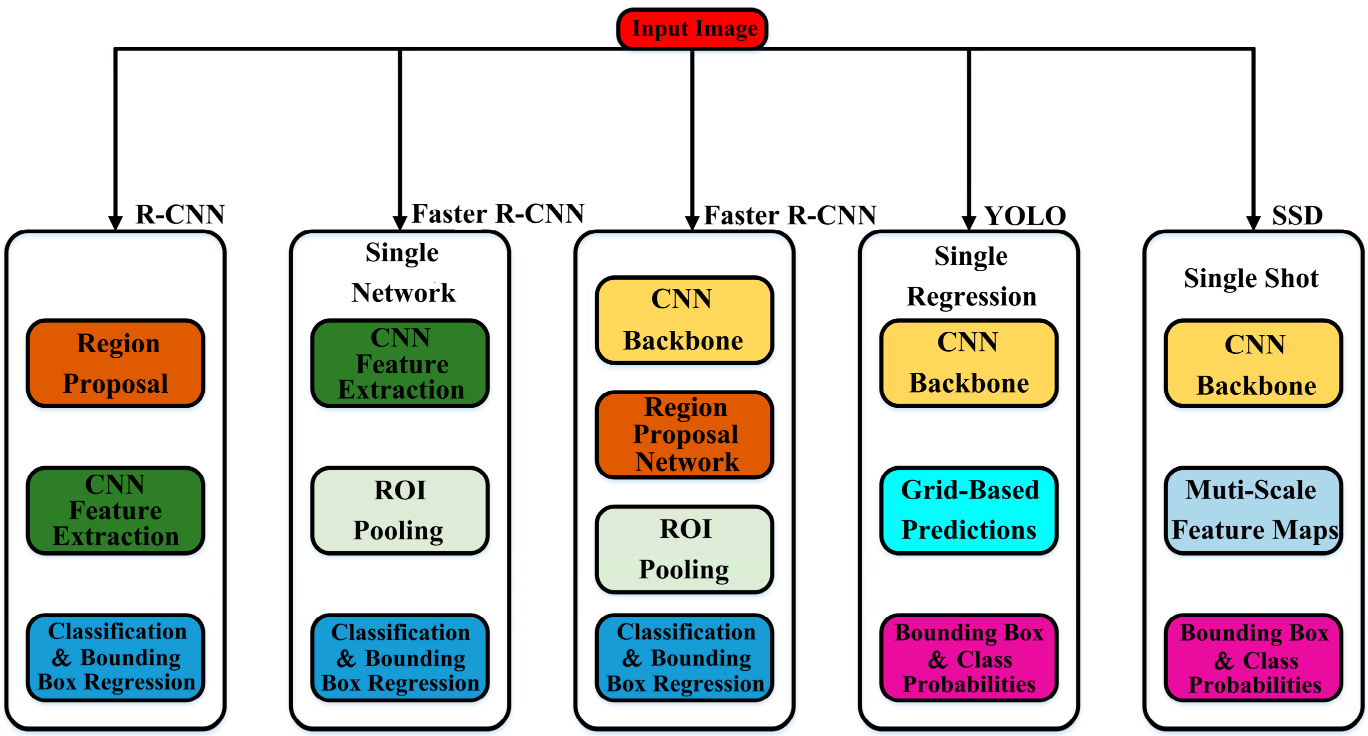


Figure 5. Comparison of typical deep learning detection algorithm architectures [33].

5.1.1. One-Stage Algorithms

One-stage detection algorithms employ a global dense prediction approach, simultaneously performing class and location predictions during the network's forward pass without the need to generate candidate regions separately. Their core advantage lies in their fast detection speed; however, due to the absence of a candidate region filtering step, their detection accuracy in complex backgrounds is slightly lower than that of two-stage algorithms. They are suitable for large-scale object detection scenarios with stringent real-time requirements. Through multiple rounds of technical iteration, single-stage algorithms have achieved a balance between accuracy and speed, giving rise to the YOLO series, Single-Shot Multi-Frame Detection (SSD), and their improved versions.

As the benchmark for single-stage algorithms, the YOLO series—since its introduction by Redmond et al. in 2015—has achieved significant performance gains through the continuous incorporation of cutting-edge optimization strategies, profoundly influencing the direction of development in the field of real-time object detection. Its underlying principles are illustrated in Figure 6. YOLOv1 pioneered the formulation of object detection as a single regression problem, establishing the paradigm of single-stage, end-to-end detection and achieving an order-of-magnitude improvement in detection speed [77]. YOLOv4 achieved significant accuracy improvements without sacrificing inference efficiency by integrating existing techniques, thereby establishing the "Bag of Freebies"and "Bag of Specials" optimization methodologies [78]. Released around the same time, YOLOv5, built on the PyTorch 1.7 framework, provided an extremely comprehensive training, validation, and deployment pipeline and quickly became the most widely used version [79]. YOLOv8 unified object detection, instance segmentation, and pose estimation within a single framework, achieving superior accuracy while maintaining high speed, marking YOLO's transition from a single-purpose detection tool to a general-purpose computer vision platform [80]. Subsequently, YOLOv9 introduced Programmable Gradient Information (PGI) and the GELAN architecture, effectively alleviating the information bottleneck in deep network training [81]; YOLOv10 achieved NMS-free end-to-end inference through a consistent dual-allocation strategy, significantly reducing inference latency [82]; YOLO11, officially released by Ultralytics, further optimized the unified multi-task framework [83]. While YOLO26 boosted CPU inference speed by 43% by removing DFL and adopting a native NMS-free design, providing an efficient solution for real-time detection at the edge [84].

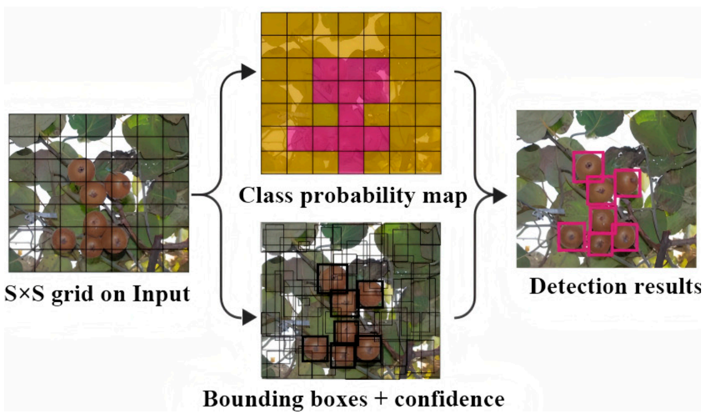


Figure 6. Principle of object detection using the YOLO algorithm [85]: The input image is divided into an S×S grid of fine lines. For each grid cell, a bounding box and its confidence score are predicted. Bounding boxes of different colors represent different object classes. In the final detection results, the colored bounding boxes represent the retained predictions, and each box is labeled with its predicted class.

As shown in Table 4, the iterative development of YOLO initially focused on addressing performance shortcomings and later gradually shifted toward balancing theoretical innovation with practical engineering applications to meet the requirements for high accuracy, high speed, and ease of deployment in multi-task scenarios such as crop monitoring, agricultural machinery navigation and automated operations, and post-harvest processing [86–95].

Table 4. Evolution of YOLO versions.

Version	Network Architecture Improvements			References
	Backbone	**Neck**	**Head**	
YOLOv1	Darknet	-	Fully connected layer for detection	[77]
YOLOv2	Darknet-19	-	Anchor box mechanism	[96]
YOLOv3	Darknet-53	FPN	Multi-scale detection heads	[97]
YOLOv4	CSPDarknet-53	PANet	Decoupled head	[78]
YOLOv5	CSPDarknet with variants n/s/m/L/x of different scales	PANet + AFF	Adaptive training anchor boxes	[79]
YOLOv6	EfficientRep	RepPAN	Decoupled head + distributed loss	[98]
YOLOv7	ELAN	PANet + MPConv	Compound scaling + guided head	[99]
YOLOv8	C2f	PAN-FPN	Decoupled detection head + anchor-free	[80]
YOLOv9	GELAN	PGI-based	GELAN-based	[81]
YOLOv10	CSPDarknet	PSA + CIB	Decoupled head	[82]
YOLOv11	C3k2	C2PSA + SPPF	Decoupled head + anchor-free	[83]
YOLOv26	End-to-End optimized	-	NMS-free native	[84]

The SSD series of algorithms was formally proposed by Liu et al. at the European Conference on Computer Vision (ECCV) conference. Featuring innovative multi-scale feature map detection, it achieves comprehensive detection of crop targets of various sizes. While maintaining end-to-end real-time inference speed, its detection accuracy surpassed that of mainstream two-stage algorithms of the time for the first time, establishing it as a mainstream single-stage algorithm that has evolved in parallel with the YOLO series. As shown in Figure 7, SSD adopts Visual Geometry Group 16 (VGG16) as the backbone network for preliminary feature extraction. Building upon this, the core innovation lies in the appended convolutional layers. These layers generate a pyramid of feature maps with progressively decreasing spatial resolutions. The detection heads are applied to multiple feature maps at different scales. This design enables the network to predict bounding boxes and class scores on coarser feature maps for large objects and to preserve more spatial details for small objects.

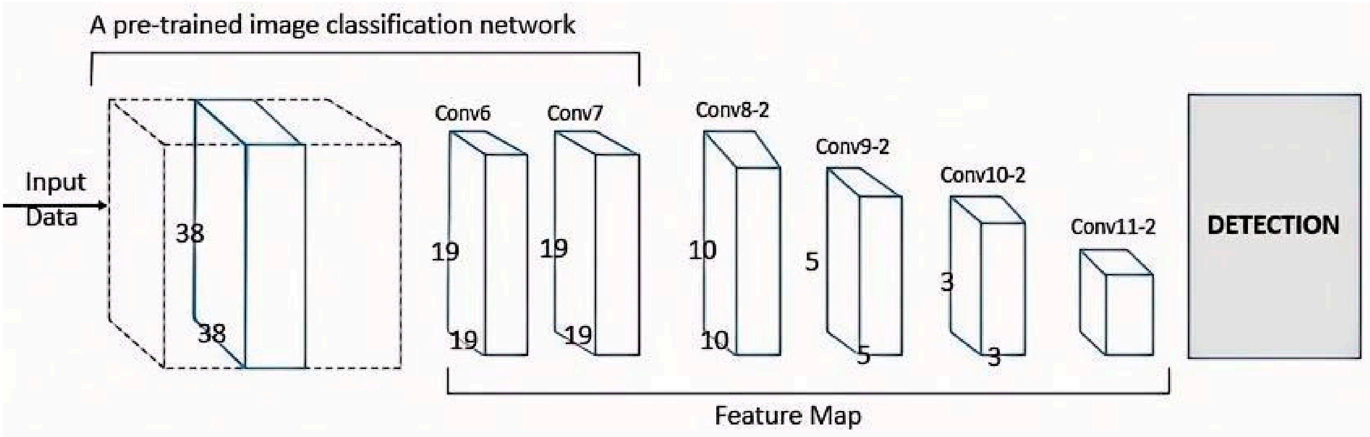


Figure 7. Network architecture of SSD [100].

5.1.2. Two-Stage Algorithms

Two-stage detection algorithms employ a phased approach involving candidate region generation and refined classification and localization. In the first stage, potential target regions are generated using a Region Proposal Network (RPN) or selective search; in the second stage, feature extraction and refined regression are performed on the candidate regions. This approach significantly reduces background interference through candidate region filtering, leading to a marked improvement in detection accuracy. However, the redundant process results in high computational complexity and relatively slow detection speeds, making it more suitable for high-value crop harvesting scenarios that demand extremely high accuracy.

R-CNN pioneered by Girshick et al. in 2014, was the first two-stage algorithm to introduce CNN technology into the field of object detection, thereby significantly improving recognition accuracy. Its process involves first generating candidate regions using selective search, extracting features from each candidate region via the AlexNet network, then using an SVM classifier for category determination, and finally optimizing positional accuracy through bounding box regression. While this algorithm represented a significant improvement over traditional machine learning algorithms, the need to process candidate regions individually resulted in severe computational redundancy, limiting detection speed to just 5 FPS [101]. In 2015, Girshick proposed the Fast R-CNN algorithm, which introduced region of interest (ROI) pooling to enable feature map sharing, boosting detection speed to 15–20 FPS and effectively addressing R-CNN's computational redundancy issue [102]. In the same year, Ren et al. proposed the Faster R-CNN algorithm, which replaced selective search with an RPN to generate candidate regions, significantly reducing generation time and boosting detection speed to 50 FPS. At the same time, the introduction of an anchor mechanism improved the quality of candidate regions, achieving a synergistic improvement in both accuracy and speed [103]. In 2017, He et al. proposed Mask R-CNN, which added a masking branch to the Faster R-CNN architecture to enable simultaneous object detection and instance segmentation. By using ROI Align technology to eliminate quantization errors in ROI pooling, it provided critical technical support for the precise segmentation of overlapping fruit [104]. Figure 8 shows the network architectures of R-CNN and its improved versions.

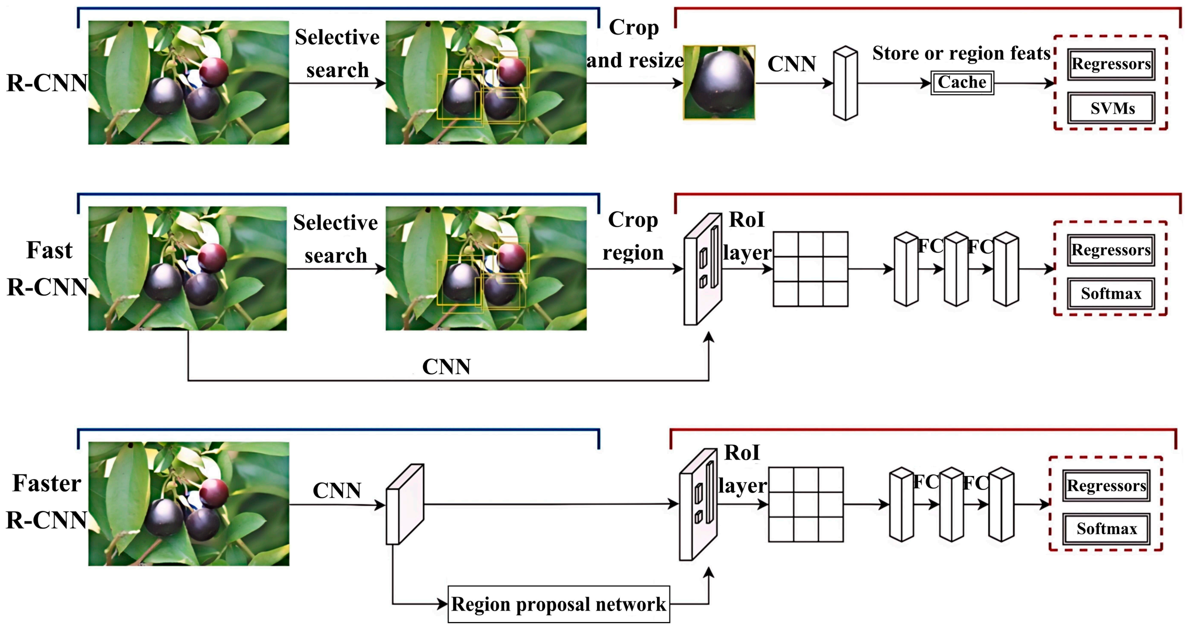


Figure 8. Network architectures of typical R-CNN, Fast R-CNN, and Faster R-CNN for fruit detection and recognition [105]: The yellow box represents the RoI extraction method; the dashed box represents the predicted head; the blue line indicates the extraction of external candidate boxes, and the red line indicates the internal feature flow and candidate box generation network.

5.2. Methods Based on the Transformer Architecture

The Transformer architecture, first proposed by Vaswani et al., is centered on the self-attention mechanism, which captures global dependencies by computing the correlation between any two positions in a sequence. Thanks to its powerful global modeling capabilities, the Transformer quickly became dominant in the field of natural language processing. Subsequently, in 2020, Carion et al. proposed DETR, marking the first application of the Transformer architecture to the field of object detection. Since then, DETR and its derivative algorithms have become the mainstream methods for this architecture in the field of object detection.

DEtection TRansformer

DETR reformulates object detection as a set prediction problem, achieving end-to-end detection through an encoder–decoder structure and a bipartite matching loss, thereby completely eliminating the reliance on manual components such as anchor box design, region proposal networks, and non-maximum suppression. However, the original DETR suffers from shortcomings such as slow training convergence and poor performance in detecting small objects. Introduced in 2021, Deformable DETR incorporated the sparse sampling concept of deformable convolutions into the attention mechanism, enabling each query to focus only on key sampling points around the reference point. This reduced the number of training epochs to one-tenth of that required by DETR and achieved significant improvements in small-object detection. In 2023, Zhang et al. proposed DETR with Improved deNoising anchOr boxes (DINO), which, by introducing contrastive denoising training and a hybrid query selection strategy, achieved 51.3 AP on COCO using a ResNet-50 backbone, demonstrating for the first time that the DETR family has the potential to outperform classical CNN detectors. The following year, Zhao et al. proposed RT-DETR. By designing an efficient encoder architecture and implementing IoU-aware query initialization, they achieved inference speeds comparable to the YOLO series while maintaining end-to-end advantages. This broke the conventional perception that Transformer-based detectors are difficult to implement in real time, providing a new viable solution for real-time end-to-end detection in agricultural harvesting scenarios.

5.3. Improvements to Deep Learning Algorithms

Although deep learning algorithms, with their deep network structures, can automatically learn and extract rich semantic features from image data and have achieved remarkable results in general object detection tasks, they still face numerous challenges when directly applied to agricultural vision scenarios. On the one hand, the accumulation of network layers causes a dramatic increase in the number of model parameters and a significant rise in computational complexity, making it difficult to meet real-time operational requirements on resource-constrained embedded platforms; on the other hand, the pre-trained parameters of existing mainstream detection models are often optimized based on publicly available general-purpose datasets, whereas agricultural scenarios exhibit significant deviations from general-purpose scenarios in terms of their characteristics; direct application leads to a substantial decline in model performance. Optimizers such as stochastic gradient descent (SGD) and adaptive moment estimation (Adam) play a significant role in deep learning training; they accelerate the model's convergence process, reduce memory requirements during training, and enable the model to achieve better generalization performance. However, these optimizers do not improve the model's accuracy or detection speed during the inference phase. Therefore, to apply general-purpose pre-trained models to specific agricultural tasks while meeting real-time requirements, targeted improvements must be made to aspects such as the network architecture and data strategies. For example, crops such as grapes and kiwifruit exhibit dense clustering and severe occlusion; fruits like strawberries and blueberries are small in size with minute features that are easily obscured. To address this, many researchers have conducted targeted algorithm improvement studies focused on three core metrics—accuracy, speed, and generalization ability—while taking into account the specific growth environment characteristics of individual crops.

5.3.1. Reducing Data Costs

In agricultural object detection tasks, the generalization performance and robustness of deep learning models depend not only on algorithmic architecture design but are fundamentally constrained by the quality of the training dataset. Unlike general-purpose object detection, fruits and vegetables—as naturally growing organisms—exhibit high spatiotemporal heterogeneity in their external characteristics depending on variety, growth stage, lighting conditions, and cultivation environment. Therefore, a dataset must contain a sufficient number of samples to cover diverse environmental conditions, thereby enabling the model to learn high-level features. However, image acquisition requires field surveys by personnel who are proficient in operating precision equipment such as industrial cameras and drones; furthermore, the collected images must be annotated with object bounding boxes before they can be used for model training. Building high-quality datasets is time-consuming, labor-intensive, and costly, leading to common shortcomings in existing datasets, such as small sample sizes, class imbalance, and scene bias. To reduce data costs and minimize the reliance of deep learning on data, techniques such as data augmentation and transfer learning are widely used in object recognition and detection tasks.

Data augmentation is divided into two categories: transformational data augmentation and generative data augmentation. Transformational data augmentation rapidly expands the quantity and diversity of training samples by applying operations such as geometric transformations, color perturbations, and blending and stitching to existing collected samples, as shown in Figure 9. Currently, augmentation strategies such as Mosaic and MixUp have become standard components of mainstream detection algorithms.

Tanimoto et al.comprehensively compared the detection performance of the YOLOv5mu, YOLOv8m, and YOLOv9c on sparse and biased datasets. The experimental results show that YOLOv5mu performs best when the number of samples is extremely low and exhibits the least accuracy fluctuation on biased datasets; however, applying data augmentation or increasing the sample size significantly improves the accuracy of the other two models, and as the dataset gradually becomes more balanced, the accuracy of all models improves [106]. Addressing the issues of high cost and limited availability of depth sensors for orchard robots, Divyanth et al. employed a Pix2Pix conditional Generative Adversarial Network (GAN) to convert single RGB images into depth maps. During training, they utilized geometric and color data augmentation techniques—such as horizontal flipping, scale scaling, brightness adjustment, and color perturbation—to enhance the model's generalization ability. This method can replace RGB-D sensors to provide depth information for orchard robots [107].

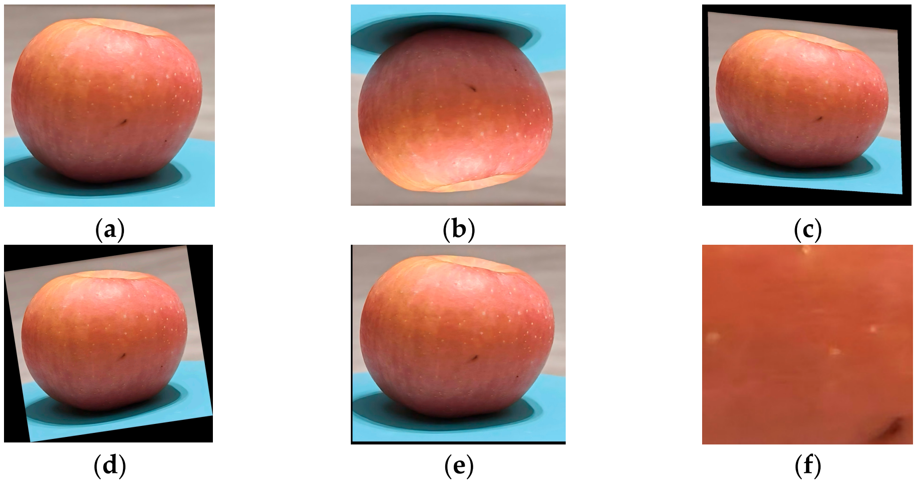
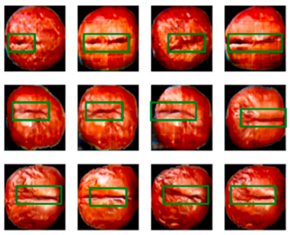
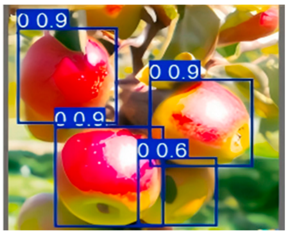
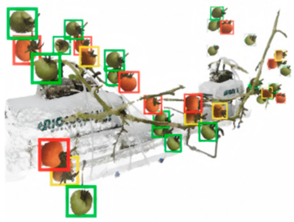


Figure 9. Data augmentation implemented for the model training [108]: (**a**) original image; (**b**) horizontal flip; (**c**) perspective transformation; (**d**) rotation; (**e**) translation; (**f**) center crop.

Generative data augmentation, exemplified by GANs and diffusion models, overcomes data limitations by generating specific scene samples missing from the original dataset, effectively alleviating issues such as class imbalance and scene bias. Table 5 lists representative studies on generative data augmentation in agricultural vision tasks; these experiments have, to varying degrees, validated the following patterns. First, an increase in data volume does not necessarily lead to improved model performance; it also depends on the degree of distribution alignment between the generated data and the target domain. If the generator only fits the limited distribution of the training set without incorporating real-world conditions such as lighting and occlusion, the performance gain from synthetic data is extremely limited. When there is a distribution bias between synthetic and real data, introducing excessive amounts of synthetic data may actually exacerbate domain drift, leading to a decline in model performance in real-world scenarios. Even when distributions align, improvements in model performance exhibit diminishing returns: while initial data augmentation can significantly improve detection accuracy, the contribution of additional data becomes extremely limited once a certain scale is reached. Second, although AI image generation technology can directly produce synthetic samples with annotations, effectively reducing the cost of manual annotation, the quality of the generated images is limited by the coverage of the training data and the precision of the prompts. Virtual engine rendering technology can also export precise annotation information simultaneously while generating images, but the visual differences between virtual scenes and real-world field environments will still affect model performance. Third, models trained solely on synthetic images exhibit significantly reduced performance in real-world scenarios and cannot match the performance levels achieved with real data. Training that combines synthetic and real data in a specific ratio yields the best results, indicating that synthetic data cannot fully replace real data. Furthermore, the proportion of synthetic data must be strictly controlled; when it exceeds a certain threshold, the model tends to learn the distribution characteristics of the generator rather than the discriminative features of real-world scenarios, thereby weakening its generalization ability.

Table 5. Comparison of generative data augmentation techniques.

Research Objective	Data Generation Method	Dataset Composition	Image Examples	Reference
To address the scarcity of nighttime pineapple detection datasets and poor detection accuracy under low visibility.	Proposed an enhanced CycleGAN, replacing the original residual blocks with ResNextBlocks in the generator.	Training set: 3000 real daytime images and 200 nighttime images, converting daytime images to nighttime; Test set: 200 real nighttime images.		[109]
To solve the problem of class imbalance in jujube quality grading and the instability of original GAN training and poor generated image quality.	Proposed an improved Jujube-GAN based on the ACGAN architecture.	1465 real images: 1000 healthy jujubes, 180 dried jujubes, 150 rotten jujubes, 135 cracked jujubes; generated images supplemented the three defective classes to 600 each;		[110]
To address the high labor cost of field data collection for orchard object detection and insufficient dataset scale and diversity	Used the DALL-E text-to-image model with detailed prompts to generate orchard apple images.	Training set: 1019 images after geometric augmentation (489 valid images); Validation set: 255 images; Test set: 93 real orchard images.		[111]
To evaluate the performance difference between datasets composed of AI-generated images and real collected datasets,	Generated synthetic images using the Generate Variations function of DALL-E2.	Three types of datasets: real dataset, AI-generated dataset, and combined dataset of real and generated images.		[112]
To solve the problem of heavy manual annotation workload, and to achieve automated dataset generation.	Constructed in Unreal Engine 5 automatically rendered images via virtual camera and exported YOLO-format annotations.	Training set: with/without cultivation bench backgrounds, different 3D model precision, 120/360/1200 images; Test set: 50 real greenhouse images.		[113]

In addition to generating target samples and expanding the dataset size, it is also feasible to enable the model to extract high-dimensional features from small datasets and converge quickly. Transfer learning leverages the general feature extraction capabilities of the model—acquired through pretraining on large-scale datasets—and applies them to object detection tasks for specific crops. The fundamental features—such as edges, textures, and shapes—learned by the model during pretraining are highly generalizable and do not become completely ineffective due to differences in target species or scenes. During training for the target task, it is sufficient to retain the weights of the pretrained model's backbone network and fine-tune the classification head and high-level feature layers using a small number of labeled target samples to achieve rapid convergence and excellent detection performance. This approach significantly reduces the demand for annotated samples in the target domain, enabling training that would otherwise require thousands or even tens of thousands of annotated samples to be completed with just a few hundred or even a few dozen samples, thereby fundamentally reducing the workload of manual annotation. Zhou et al. compared models such as YOLOv8x and Fast R-CNN and, through cross-species transfer learning, transferred model weights pre-trained on a public apple dataset to the camellia fruit detection task. Using only 100–200 annotated samples, they achieved detection accuracy close to that of the full dataset; among these, YOLOv8x, which performed best, reached an mAP50 of 94–95% [114]. Chen et al. adopted a two-stage transfer strategy—first freezing the skeleton network to train the classification head, then unfreezing the entire network for fine-tuning—combined with image scaling and loss function optimization, achieving a recognition accuracy of 97.17% for Chinese cabbage at different growth stages [115]. Zhao et al. slightly adjusted the pre-trained YOLOv8s weights on the COCO dataset and applied them to detect apples on fruit trees and on the ground, achieving an mAP50–95 of 72.6% [116]. Wang et al. independently designed a category-based automatic data balancing algorithm. They first froze the head network of the MobileNetV3 backbone during training and then unfroze the entire network for fine-tuning, resulting in a model mAP of 88.56%, with only 17.92% of the parameters of the original YOLOv4 and a 112% improvement in detection speed [117]. Ufuah et al. used Style-GAN2 ADA to synthesize images to expand the training set and performed transfer learning on a DenseNet network, achieving fast and accurate classification of three types of dates [118]. Espejo-Garcia et al. determined the optimal generator parameters for the DCGAN network and verified that a model initialized with ImageNet pre-trained weights, fine-tuned on an agricultural dataset, and trained using a GAN-augmented dataset achieved superior performance; the Xception network used for detection achieved accuracy rates of 99.07% and 93.23% on a standard test set and a noisy test set, respectively [119].

Although all studies used independent test sets to validate model performance, these test sets typically originated from the same distribution as the training data—the same orchards, the same crop varieties, and similar imaging conditions. Only a few studies incorporated domain adaptation techniques; for example, CycleGAN—an unsupervised domain adaptation method—generates images in the target domain while performing a reverse transformation to verify their plausibility. Wu et al. used this method to convert daytime images from the training set into nighttime images, thereby bridging the gap between the training and target domains. Furthermore, while most studies utilized real-world images captured in the field or images collected during harvesting robot operations, the scale of validation remained limited, and none extended to field-scale testing under actual harvesting conditions. Therefore, although the aforementioned data augmentation and transfer learning methods reduce the cost of data acquisition or lessen the reliance of deep learning algorithms on data to some extent, their effectiveness in real-world harvesting scenarios still lacks sufficient empirical support.

https://doi.org/10.3390/agriculture16171924

5.3.2. Model Compression and Acceleration

Unlike industrial applications, agricultural harvesting robots are typically small mobile platforms, which impose stricter requirements on device size, weight, and energy consumption. Model deployment is subject to multiple constraints, including hardware computing power, memory capacity, and battery life. Large-scale deep learning models with a high number of parameters not only make real-time inference difficult but also significantly increase the overall manufacturing costs and power consumption of the robot, making them unsuitable for long periods of continuous operation in the field. Therefore, model lightweighting is a critical step in transitioning target recognition technology from the laboratory to practical application. It involves reducing the number of parameters and computational load while minimizing accuracy loss, and optimizing the inference process to ensure stable operation on low-power edge devices, thereby balancing accuracy, speed, and deployment costs. As shown in Table 6, most existing studies use mainstream single-stage detection algorithms as baselines for improvement. By employing techniques such as network replacement, the introduction of lightweight modules, and pruning to optimize the network structures of the backbone, neck, and head components, these studies have developed various lightweight models suitable for agricultural robots.

Most studies in the table include the number of parameters, computational complexity, and their reduction rates as evaluation metrics; some studies also evaluate model size, inference speed on different endpoints, and the time required for inference on a single frame. Among them, YOLO-TomatoSeg is the model with the greatest compression rate across all studies, reducing model size, the number of parameters, and computational complexity by 68.3%, 38.9%, and 33.3%, respectively, with a model size of only 2.52 MB. In contrast, Seedling-YOLO achieved the lowest compression rate, with parameter counts and computational complexity reduced by only 20% and 16.5%, respectively; however, its mAP0.5 improved by 4%, representing the most significant increase in accuracy. The wide variation in compression levels among these models stems from the need to adjust compression methods and balancing strategies based on differences in detection targets and experimental conditions. The specifications of the baseline models used also influence the degree of compression; models with more redundant feature layers offer greater potential for compression. As the underlying feature extraction network, the backbone typically accounts for the largest share of parameters and computational load in the entire network; replacing it entirely with a lightweight backbone can significantly reduce the number of parameters. Redesigning efficient convolutional operators to replace standard convolutions can limit channel interactions and reduce redundant convolutional computations. Building on this, attention modules are introduced to compensate for the decline in feature expressiveness caused by backbone simplification, and feature fusion structures are optimized to further enhance feature utilization efficiency. Some studies have also improved anchor box generation strategies to adapt to the scale distributions of specific targets. The combination of these approaches enables most models to maintain accuracy while reducing the number of parameters; some works have even achieved simultaneous optimization of accuracy and lightweight design. Although the aforementioned studies have achieved significant results in the lightweight design of network architectures, none of the models have undergone full integration and field deployment validation on actual harvesting robot platforms. As indicated by the validation conditions summarized in the table, these studies primarily conducted offline evaluations using images from fixed test sets. Some studies conducted inference speed tests using only CPUs or edge computing devices to simulate the resource-constrained environment of harvesting robots and reported high frame rates; however, they did not address the validation and evaluation of quantized models, nor did they include integration testing with the robot's operating system or robustness verification for long-term field operations.

Table 6. Comparison of model compression and acceleration techniques.

Research Objective	Improvement	Validation Conditions	Performance	Reference
Lightweight YOLO11n for tomato segmentation, detection and yield estimation	C3k2-F Module Conv → DSConv GIoU	242 3D images and 9 videos captured by a depth camera in a greenhouse; Jetson Orin NX	mAP50: 93.3% Params: 2.66 M GFLOPs: 8.0 FPS: 62	[120]
YOLO-TomatoSeg for tomato recognition	CSPDarkNet → ShuffleNetV2 SE	754 point cloud images captured by two industrial cameras in a greenhouse; Intel i9-12900H CPU, NVIDIA RTX 3060	AP: 99.01% Model Size: 2.52 MB FPS: 70.0 Params: 1.1 M GFLOPs: 4.6	[121]
YOLOv9s-Pear for small-object recognition of pears	AConv → SCDown RepNCSPELAN4 → C2FUIBELAN YOLOv10 Detection Head	158 RGB images taken with a cell phone in natural settings; Intel (R) Xeon (R) Gold 5318Y CPU @ 2.10 GH, NVIDIA A16 GPU	mAP50: 99.1% mAP50-95: 84.8% Params: 4.66 M GFLOPs: 19.8	[122]
YOLO-GS for aquatic vegetable recognition and detection	Conv → GhostConv, C3-GS Module 160 × 160 Small-Target Detection Head Focal EIoU	150 images captured by UAV under natural light; Industrial control computers without graphics card	mAP50: 95.7% Params: 3.75 M GFLOPs: 9.5 CPU FPS: 28.7	[123]
YOLOGPnet for grape detection	C2f → SENetV2 SPPF → SPPELAN Focaler-IoU	236 RGB images taken with a smartphone under different lighting conditions and distances;	mAP50: 96.8% GFLOPs: 6.8 Params: 2.1 M Model Size: 4.36 MB FPS:80	[124]
Seedling-YOLO for quality detection of broccoli field transplanting	ELAN_P CARAFE Upsampling Operator CA	600 field images captured by depth cameras and smartphones; field validation of the mobile vision platform; Intel i7-13700KF CPU, RTX 4080 GPU,	mAP50: 94.3% Params: 4.98 M GFLOPs: 11.6 FPS: 29.7	[125]
YOLOv8n-seg for instance segmentation and plant height measurement of lettuce	CSPDarkNet → FasterNet Restructuring the Channel Dimension	220 images captured by an RGB-D camera in a greenhouse; AMD Ryzen 7700 Processor RTX 2070 Graphics Card	mAP50: 99.5% FPS: 25 Params: 1.369 M	[126]
LBDC-YOLO for broccoli detection	Slim-Neck (Conv → GSConv) Triplet Attention PAN-FPN → BiFPN	84 images captured with a depth camera and a cell phone in a field setting; Intel Xeon Silver 4110 2.1 GHz CPU NVIDIA Quadro P4000 GPU	AP50-95: 94.44% Params: 1.92 M GFLOPs: 6.5 Model Size: 3.5 MB FPS: 23.0	[127]
Light-YOLO-Pepper for pepper plug seedling detection	Conv → DSConv CSPDarknet → CSP-MobileNet SE	864 images taken with a cell phone in a greenhouse with stable lighting; Intel Core i9-13900K, NVIDIA RTX 4090	mAP50: 93.6% Params: 1.82 M GFLOPs: 3.2 GPU/CPU FPS: 168.4/28.9	[128]

As shown in Figure 10, Knowledge Distillation (KD) transfers knowledge from a larger teacher network to a lightweight student network, enabling the model to achieve superior detection performance while maintaining its current number of parameters and inference speed, thereby effectively addressing the issue of low accuracy in lightweight detection models within complex orchard environments [129].

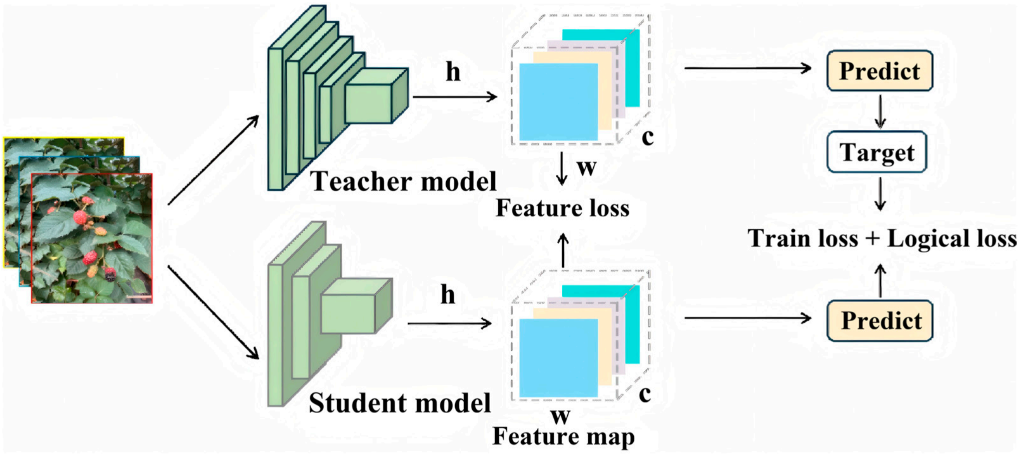


Figure 10. Schematic diagram of knowledge distillation [130].

For example, Leng et al. developed the SMLO-YOLO lightweight model for strawberry detection based on YOLOv11s, employing the Block-Correlation Knowledge Distillation (BCKD) boundary and category-aware knowledge distillation strategy, combined with the HDP-Neck cross-scale fusion architecture and ShapeIoU shape-aware loss. The distilled model has only 6.49M parameters and 15.0 GFLOPs, achieving an mAP50 of 92.4%—a 2.9% improvement over the original YOLOv11s—with an inference speed of 256.41 FPS and a 31% reduction in parameters [129]. Lingamgunta et al. constructed a multi-granularity knowledge distillation framework, using DeiT-Base as the teacher model to distill the lightweight DeiT-Tiny. The student model has approximately 5.28M parameters and a single-image inference latency of 8.79 ms; the ensemble model achieved a grading accuracy of 90% for cashew apples [131]. Li et al. proposed the D^3-YOLOv10 lightweight detection framework, which employs a FreeKD knowledge distillation scheme based on semantic frequency hints. Using an improved YOLOv10s as the teacher model to distill YOLOv10n, the final model has 3.72M parameters and 8.6 GFLOPs, representing reductions of 54.0% and 64.9%, respectively, compared to YOLOv10s, while achieving an mAP0.5 of 91.8% and an inference speed of 80.1 FPS [132].

5.3.3. Robustness Improvement in Complex Scenarios

Agricultural production environments are highly unstructured. In open-field cultivation scenarios, light intensity fluctuates dramatically over time and with weather conditions. Fruit is commonly obscured by branches and leaves and overlaps with one another; small fruits have faint features, and key harvesting points—such as fruit stems—are extremely small. These factors can lead to missed detections and false positives in the model, severely affecting the operational reliability of harvesting robots. Improving a model's robustness in complex scenarios primarily involves enhancing its ability to extract target features, suppressing background interference, and simultaneously improving its adaptability to multi-scale targets and extreme environments. Table 7 summarizes and analyzes the optimization methods for deep learning models in the YOLO and R-CNN families, focusing on enhancements to attention mechanisms, The arrows indicate the performance improvement of the improved model compared to the baseline model. It also presents the backbone networks and final performance results of the optimized models in complex scenarios involving various target crops. Some studies have also achieved significant improvements in model performance by refining non-maximum suppression and loss functions.

Table 7. Comparison of model performance improvement studies in complex scenarios.

Object	Model	Improvement	Validation Conditions	Performance	References
Apple	Mask R-CNN	DANet [1]	219 images taken with a cell phone in the orchard; Intel Core i7-7700HQ, NVIDIA GTX 1070	bbox_mAP: 93.2% mask_mAP: 92.0% F1 score: 96.9% FPS: 3.7	[133]
	Mask R-CNN	SE-CBAM, RANSAC-LM sphere fitting	1000 RGB and depth images captured by a depth camera in orchard, Indoor Simulation Experiments, Field Trials; Intel i7-9700 CPU, NVIDIA RTX 2060	Mask IoU:0.89/0.82/0.65	[134]
	YOLOX-Tiny	CBAM, ASFF	30 images taken in an orchard during bagging and under artificial nighttime lighting; GTX1650 GPU, Jetson Nano	AP: 96.76% F1 score: 0.95 Params: 5.4 M FPS: 65/26.3	[17]
	YOLOv5s	EfficientFormer Soft-NMS	50 Complex Orchard xImages, Indoor Simulated Harvesting Experiment; GTX1080ti	AP: 98.84% (4.3% ↑) Params: 10.78 M FPS: 28	[135]
	YOLOv4	DSC CIoU	1038 images from an orchard depicting complex scenarios such as fruit bagging and nighttime conditions; NVIDIA GTX1080Ti GPU	AP: 93.42% F1 score: 0.9035 Model Size: 29.8 M FPS: 63.2	[136]
Tomato	YOLO-MSRF	MS-CPAM [2] C2f-DCB	A total of 208 RGB and NIR images captured under various lighting conditions; NVIDIA RTX 3090 GPU	mAP50: 2.3% ↑ Params: 4.1 M GFLOPs: 14.6 FPS: 105.07	[137]
	Faster R-CNN	Soft-NMS [3] K-means anchor box clustering	1786 images captured by a cell phone inside the greenhouse; Intel Core i7-7700K, NVIDIA GTX1080Ti	mAP: 90.7% (5.5% ↑) Model Size: 115.9 M FPS: 13.7	[138]
Strawberry	YOLOv8s	GAM [4] small-object detection head	600 images taken with a depth camera in an orchard, depicting different growth stages and environmental conditions; Intel Xeon Gold 6248 CPU, NVIDIA Tesla V100 GPU	mAP:91.5% (4.6% ↑) FPS: 53	[139]
	YOLOv8s	CBAM [5]	Images of strawberries at different times and growth stages; AMD EPYC 7453 CPU, NVIDIA RTX 4090 GPU	AP:86.2% FPS: 32.7	[140]

Table 7. *Cont.*

Object	Model	Improvement	Validation Conditions	Performance	References
Grape	Mask R-CNN	ECA [6] DUC [7] upsampling	138 images taken with a camera in the vineyard; NVIDIA RTX 3090	detection AP: 60.1% (1.4% ↑) segmentationAP: 59.5% (1.6% ↑) FPS: 16	[141]
Citrus	Mask R-CNN	CBAM Soft-NMS	200 images captured by a dual-lens camera in an orchard; Intel Xeon 3204, NVIDIA RTX 2080 Ti	P: 95.04% R: 88.11% F1:91.44%	[142]
Persimmon	Faster R-CNN	ECA Serial Skip-Layer Multiscale Fusion K-means anchor box clustering	930 images taken under various lighting conditions and at different distances; Intel Core i7-12700 CPU, NVIDIA RTX 3060	mAP:98.4% (11.8% ↑) FPS: 1.39	[143]

[1] Dual Attention Network; [2] Multi-Scale Channel Position Attention Module; [3] Non-Maximum Suppression; [4] Global Attention Mechanism; [5] Convolutional Block Attention Module; [6] Efficient Channel Attention; [7] Dense Upsampling Convolution.

The core principle of attention mechanisms lies in enabling the network to automatically focus on key semantic features in the target region through learnable weight allocation, while suppressing unstructured distractions such as the background, branches, leaves, and lighting. In terms of specific implementation, attention modules such as CBAM, ECA, GAM, and SE are embedded into the backbone network or mask heads to enhance feature representations at multiple levels. Particularly in extreme scenarios such as heavy occlusion, low light, and similar-colored backgrounds, attention mechanisms can effectively mitigate the problem of feature loss. As shown in Figure 11, SE, as the earliest channel attention mechanism, compresses spatial information into a one-dimensional channel description vector via global average pooling, then uses two fully connected layers to learn the activation weights for each channel; however, this dimensionality reduction prevents the model from capturing inter-channel relationships, resulting in information loss. Building on this, ECA removes the dimension-reduction fully connected layers and instead uses one-dimensional fast convolutions to directly perform local cross-channel interactions on the channel vectors after global pooling.

The CBAM and GAM attention mechanisms employ two serial network branch structures (Figure 12): CBAM performs global average pooling and max pooling on its spatial branch, generating a spatial attention map after convolutional fusion; GAM uses dilated convolutions to expand the receptive field and capture global contextual information. In contrast, DANet employs parallel network branches and uses self-attention mechanisms to construct dense dependency matrices separately in the spatial and channel dimensions (Figure 12). Although this results in a relatively rapid increase in computational complexity, it offers significant advantages for segmentation tasks characterized by severe overlap and highly variable shapes. This type of spatial-channel attention mechanism enables the model to simultaneously focus on both the spatial and channel information of an image, making it more suitable for scenarios where the scale of objects varies significantly. In terms of model performance, after introducing the attention mechanism, the average accuracy of most models in the table reached over 90%, representing a significant improvement over baseline models. This confirms that the ability of the attention mechanism to adaptively enhance specific regions is a key factor in achieving accuracy breakthroughs in complex agricultural scenarios.

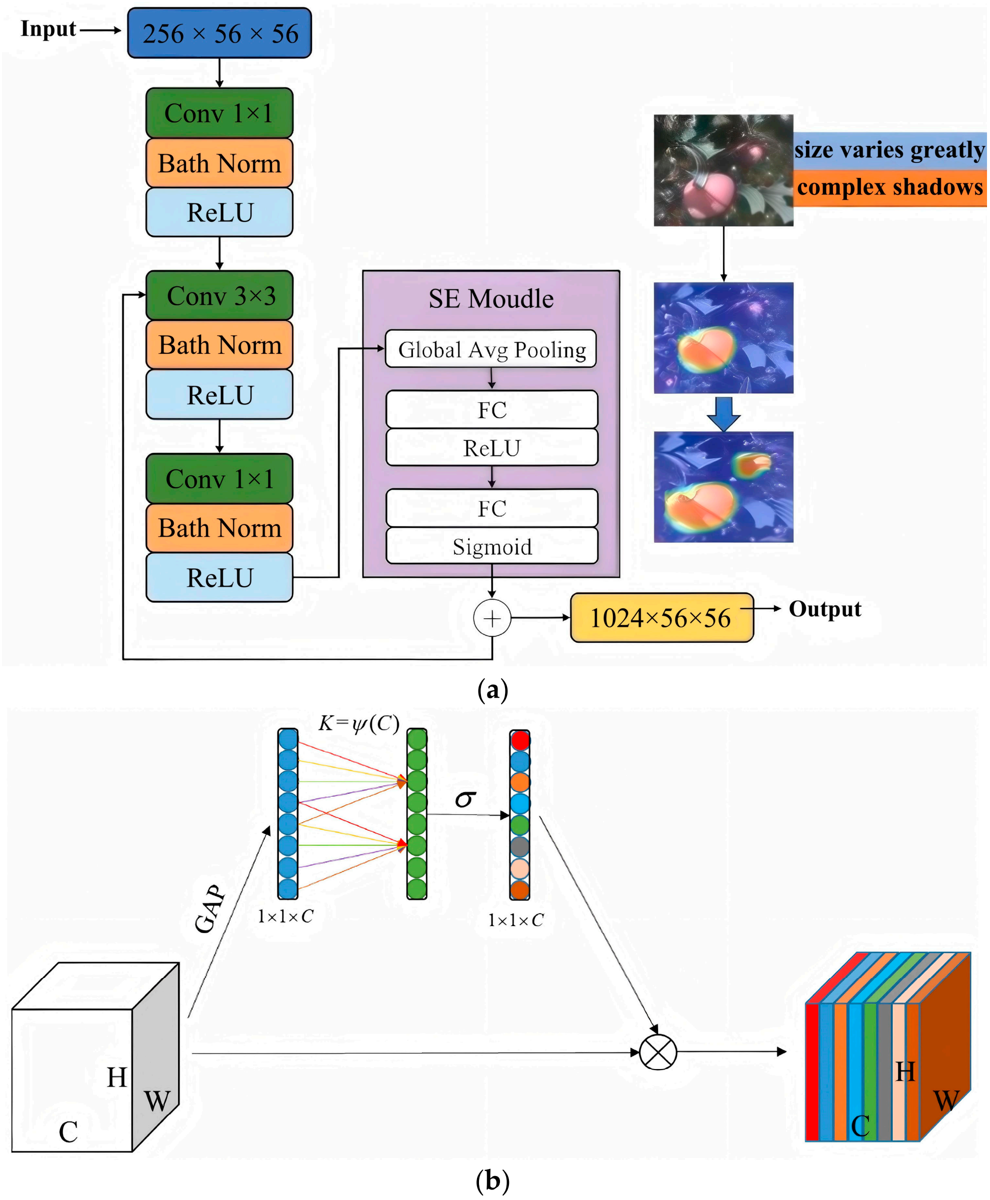


Figure 11. Network architecture of SE and ECA [134,141]: (**a**) SE; (**b**) ECA The different colors in the figure represent different feature channels, and the colored vectors on the right indicate the attention weights calculated for each channel, which are used to apply a weighting to the original feature map.

To address the false negatives caused by traditional NMS—which, due to its rigid threshold, removes adjacent high-confidence candidate boxes in densely overlapping scenarios—NMS optimization employs a soft-NMS strategy. This approach continuously attenuates the confidence scores of overlapping boxes based on the IoU rather than setting them to zero outright. Consequently, it suppresses redundant boxes while preserving genuinely occluded targets, enabling the model to maintain high recall even in typical occlusion scenarios such as fruit adhesion and interlaced branches and leaves, while still maintaining a high recall rate.

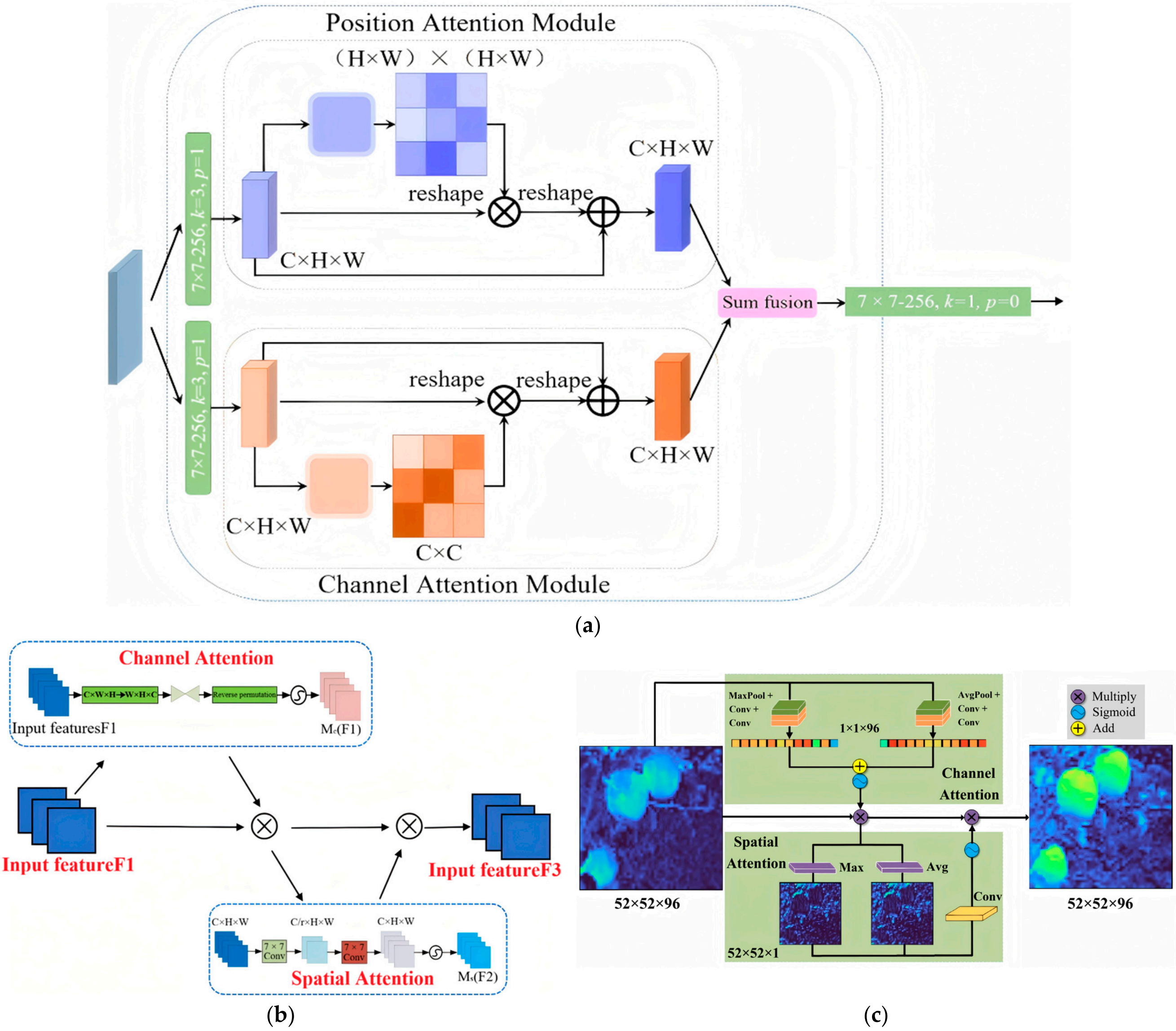


Figure 12. Network architecture of DANet, GAM and CBAM [17,133,139]: (**a**) DANet; (**b**) GAM; (**c**) CBAM.

6. Summary and Outlook

6.1. Summary

Global demand for fruits and vegetables continues to grow, yet harvesting—the most labor-intensive stage of agricultural production and the one with the highest proportion of labor costs—has long faced challenges of low operational efficiency and significant wastage. Agricultural harvesting robots are the core technological solution to this problem, and their operational capabilities rely heavily on the accuracy and speed of target recognition technology: the recognition system must distinguish fruits and vegetables from their background in complex orchard environments, obtain their 3D pose, and rapidly assess their ripeness to provide perceptual support for the actuators. To address this need, this paper systematically reviews the development trajectories and latest advancements of three categories of recognition technologies: digital image processing, traditional machine learning, and deep learning. The main contributions include: summarizing the fundamental principles, evolutionary paths, and applicability boundaries of these three technologies; elucidating their essential differences at the feature extraction level—namely, the progression from manually defined rule-based features, through data-driven shallow learning, to end-to-end deep abstraction; summarizing the optimization strategies for each technology in complex agricultural scenarios, and clarifying their performance limits and applicable scenarios under different operating conditions; and comparing the strengths and weaknesses of the three approaches across multiple dimensions—including recognition accuracy, real-time performance, deployment costs, data dependency, and scenario robustness—to provide a reference for subsequent algorithm selection and improvement.

Digital image processing relies on manually designed, rule-based features such as color, shape, and texture. It requires minimal computational resources and offers good real-time performance, making it suitable for controlled-environment agriculture and standardized orchards where there is a significant contrast between targets and backgrounds and lighting conditions are stable. However, due to the limited expressive power of manually designed features, even with multi-feature fusion, it still suffers from poor cross-scenario transferability. Accuracy drops significantly under conditions of severe occlusion, fruit overlap, or similar-colored backgrounds. Furthermore, parameters require manual tuning, and performance is clearly capped by the limitations of the rule-based framework.

Traditional machine learning, utilizing classifiers such as SVMs and random forests, facilitates a shift from rule-driven to data-driven approaches. It offers low training costs and strong interpretability for small-sample tasks such as crop variety identification and ripeness grading. However, the upper limit of model performance is determined by feature engineering, making it prone to the curse of dimensionality and overfitting. Shallow architectures struggle to capture highly nonlinear relationships, resulting in insufficient generalization capabilities in orchards with severe occlusion and drastic fluctuations in lighting conditions. Although dimension reduction and ensemble learning can mitigate these shortcomings, this technology is primarily used as a supplement to deep learning, handling lightweight, specialized tasks on edge devices with limited computing power.

Deep learning has become the mainstream solution for fruit and vegetable recognition due to its end-to-end automatic feature extraction capabilities. Technological evolution has progressed from the CNN paradigm, represented by YOLO and Faster-R-CNN, to Transformer-based detection architectures such as Deformable-DETR and RT-DETRv2. This field still faces challenges such as high data annotation costs, the difficulty of balancing accuracy with model lightweightness, and insufficient robustness in complex field scenarios. Existing research has established a comprehensive optimization framework: data augmentation combined with transfer learning reduces reliance on labeled data; techniques such as backbone network replacement, efficient convolutional operators, network structure reparameterization, and parameter compression achieve model lightweighting while minimizing accuracy loss, making the models suitable for real-time inference on edge devices; feature enhancement and multiscale detection optimization improve recognition performance under occlusion, for small objects, and under extreme lighting conditions; and the synergistic use of these various improvement strategies enables a balance among model recognition accuracy, inference speed, and deployment costs.

6.2. Outlook

With the rapid development of smart agriculture and the continuing global shortage of agricultural labor, the automation of fruit and vegetable harvesting has become an inevitable trend in industrial upgrading. As the core perceptual foundation for harvesting robots, target recognition technology is evolving from simple fruit detection and localization toward multidimensional intelligent perception, and shifting from crop-specific models to generalized systems. Future technological innovations will closely align with the practical needs of agricultural production. Through dual breakthroughs in algorithmic architecture and application models, these innovations will drive the large-scale commercial deployment of harvesting robots.

(1) Deep integration of multimodal features: Overcoming the limitations of single-modality RGB visual perception, this approach deeply integrates information from multi-modal sensors such as near-infrared, depth, thermal imaging, and LiDAR. By leveraging the complementary strengths of different modalities—including robustness to lighting changes, acquisition of geometric information, and perception of material properties—it systematically addresses challenges such as sudden changes in lighting, interference from similar-colored backgrounds, and target reconstruction under foliage occlusion. Through the collaborative extraction of multimodal features and integrated decision-making, the stability and reliability of the perception system in complex field environments are enhanced, providing harvesting robots with more comprehensive and precise environmental and target information.

(2) Research on algorithm application and deployment: Currently, most fruit and vegetable recognition algorithms remain at the stage of laboratory simulation and dataset validation, with a significant gap remaining before they can be deployed on harvesting robots for stable operation. In the future, model deployment should be a key research focus. Given the resource constraints of harvesting robots—such as limited computing power and memory—after completing model lightweighting, efforts should be directed toward deployment-level tasks such as model quantization, pruning, and inference acceleration. This will ensure that trained algorithmic models maintain recognition accuracy and inference speed after being ported to embedded hardware, thereby narrowing the performance gap between controlled laboratory environments and actual field operations.

(3) Development of self-supervised and weakly supervised learning technologies: Future research should fully leverage the value of unlabeled agricultural images collected in the field, focusing on self-supervised pre-training techniques such as contrastive learning and masked image modeling. This will enable models to autonomously learn general visual features from agricultural images, thereby fundamentally reducing reliance on large-scale, finely annotated data. Combined with weakly supervised and semi-supervised learning techniques, rapid model training and optimization can be achieved with only a small amount of simple annotation. This effectively addresses the high costs, lengthy cycles, and uneven sample distribution associated with building agricultural datasets, thereby enhancing the model's generalization ability and adaptability across different regions and cultivation practices.

(4) Integration of fruit recognition algorithms with picking point localization: In the future, efforts should focus on developing a multi-task integrated network that combines detection, segmentation, and regression of key picking points, deeply integrating RGB image data with depth point cloud information to simultaneously output fruit class, mask, key picking points, and 3D spatial pose. To address complex scenarios such as occlusion and small targets, the learning of spatial geometric relationships by the network should be enhanced to provide the robotic arm's end-effector with perception outputs that can be directly used for grasping decisions.

7. Conclusions

This study provides a systematic review of fruit and vegetable target recognition technologies used in harvesting robots, covering a variety of algorithms from digital image processing, traditional machine learning, and deep learning, and analyzing the implementation principles, evolutionary history, and applicable scenarios of these three paradigms. Deep learning, with its end-to-end automatic feature extraction capabilities, has become the dominant solution in contemporary complex agricultural scenarios. Crucially, this study also evaluates these technologies based on the practical operational requirements of harvesting tasks, rather than relying solely on general detection metrics. At the same time, it clearly identifies long-term bottlenecks hindering large-scale adoption, including high data annotation costs, difficulties in edge deployment, and insufficient robustness under conditions of severe occlusion and drastic changes in lighting. Although downstream tasks such as sorting and placement present further physical execution challenges, visual recognition systems remain the fundamental prerequisite for fully autonomous harvesting robots. The optimization strategies summarized in this study provide a solid theoretical foundation for future engineering implementations. Addressing deployment bottlenecks on resource-constrained embedded platforms, combined with technologies such as multimodal fusion and semi-supervised learning, will be the key path to the widespread adoption of intelligent agricultural equipment in the future.

Author Contributions: Conceptualization, D.H. and X.C.; methodology, S.X.; software, S.X.; validation, D.H., Q.Z. and S.Z.; formal analysis, S.Z.; investigation, S.X. and S.Z.; resources, Q.Z. and X.C.; data curation, S.X.; writing—original draft preparation, D.H. and S.X.; writing—review and editing, D.H., S.Z. and Q.Z.; visualization, S.X.; supervision, D.H.; project administration, D.H.; funding acquisition, D.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 52105304), the Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions (No. PAPD-2023–87), the 2023 Open Project of Key Laboratory of Modern Agricultural Equipment and Technology (Jiangsu University), Ministry of Education and High-tech Key Laboratory of Agricultural Equipment and Intelligence of Jiangsu Province (No. MAET202327), and the Jiangsu University-Level Research Project (No. 25A003).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. FAO. Unlocking the Full Potential of Fruits and Vegetables for the Transformation of Agrifood Systems. Available online: https://www.fao.org/events/detail/unlocking-the-full-potential-of-fruits-and-vegetables-for-the-transformation-of-agrifood-systems/en (accessed on 21 July 2026).
2. FAO. Production Quantity of the Furits and Vegetables in the World. Available online: https://www.fao.org/faostat/zh/#data/.QV (accessed on 21 July 2026).
3. Gao, W.; Liu, J.; Deng, J.; Jiang, Y.; Jin, Y. Research Status and Trends in Universal Robotic Picking End-Effectors for Various Fruits. *Agronomy* **2025**, *15*, 2283. [CrossRef]
4. Chen, K.; Li, T.; Yan, T.; Xie, F.; Feng, Q.; Zhu, Q.; Zhao, C. A Soft Gripper Design for Apple Harvesting with Force Feedback and Fruit Slip Detection. *Agriculture* **2022**, *12*, 1802. [CrossRef]
5. Faheem, M.; Liu, J.; Chang, G.; Abbas, I.; Xie, B.; Shan, Z.; Yang, K. Experimental Research on Grape Cluster Vibration Signals during Transportation and Placing for Harvest and Post-Harvest Handling. *Agriculture* **2021**, *11*, 902. [CrossRef]
6. Xu, B.; Liu, J.; Jin, Y.; Yang, K.; Zhao, S.; Peng, Y. Vibration-Collision Coupling Modeling in Grape Clusters for Non-Damage Harvesting Operations. *Agriculture* **2025**, *15*, 154. [CrossRef]
7. Zhang, Q.; Zhao, Z.; Gao, G. Fuzzy Comprehensive Evaluation for Grasping Prioritization of Stacked Fruit Clusters Based on Relative Hierarchy Factor Set. *Agronomy* **2022**, *12*, 663. [CrossRef]
8. Liu, J.; Liang, J.; Zhao, S.; Jiang, Y.; Wang, J.; Jin, Y. Design of a Virtual Multi-Interaction Operation System for Hand-Eye Coordination of Grape Harvesting Robots. *Agronomy* **2023**, *13*, 829. [CrossRef]
9. Faheem, M.; Liu, J.; Chang, G.; Ahmad, I.; Peng, Y. Hanging force analysis for realizing low vibration of grape clusters during speedy robotic post-harvest handling. *Int. J. Agric. Biol. Eng.* **2021**, *14*, 62–71. [CrossRef]
10. Liu, J.; Yuan, Y.; Gao, Y.; Tang, S.; Li, Z. Virtual Model of Grip-and-Cut Picking for Simulation of Vibration and Falling of Grape Clusters. *Trans. ASABE* **2019**, *62*, 603–614. [CrossRef]

11. Xie, F.; Guo, Z.; Li, T.; Feng, Q.; Zhao, C. Dynamic Task Planning for Multi-Arm Harvesting Robots Under Multiple Constraints Using Deep Reinforcement Learning. *Horticulturae* **2025**, *11*, 88. [CrossRef]
12. Yang, Q.; Zhong, X.; Qu, G.; Liu, L.; Yang, X.; Hu, X.; Min, A. A Three-Finger End-Effector with retractable suction for cluster tomato harvesting: Design and dynamic performance evaluation. *Trans. ASABE* **2025**, *68*, 603–616. [CrossRef]
13. Jin, Y.; Liu, J.; Xu, Z.; Yuan, S.; Li, P.; Wang, J. Development status and trend of agricultural robot technology. *Int. J. Agric. Biol. Eng.* **2021**, *14*, 1–19. [CrossRef]
14. Xu, Z.; Liu, J.; Wang, J.; Cai, L.; Jin, Y.; Zhao, S.; Xie, B. Realtime Picking Point Decision Algorithm of Trellis Grape for High-Speed Robotic Cut-and-Catch Harvesting. *Agronomy* **2023**, *13*, 1618. [CrossRef]
15. Lei, X.; Liu, J.; Jiang, H.; Xu, B.; Jin, Y.; Gao, J. Design and Testing of a Four-Arm Multi-Joint Apple Harvesting Robot Based on Singularity Analysis. *Agronomy* **2025**, *15*, 1446. [CrossRef]
16. Wang, C.L.; Li, C.J.; Han, Q.Y.; Wu, F.Y.; Zou, X.J. A Performance Analysis of a Litchi Picking Robot System for Actively Removing Obstructions, Using an Artificial Intelligence Algorithm. *Agronomy* **2023**, *13*, 2795. [CrossRef]
17. Ji, W.; Pan, Y.; Xu, B.; Wang, J.C. A Real-Time Apple Targets Detection Method for Picking Robot Based on ShufflenetV2-YOLOX. *Agriculture* **2022**, *12*, 856. [CrossRef]
18. Yu, Y.; Xie, H.; Zhang, K.; Wang, Y.; Li, Y.; Zhou, J.; Xu, L. Design, Development, Integration, and Field Evaluation of a Ridge-Planting Strawberry Harvesting Robot. *Agriculture* **2024**, *14*, 2126. [CrossRef]
19. Guan, X.; Shi, L.; Ge, H.; Ding, Y.; Nie, S. Development, Design, and Improvement of an Intelligent Harvesting System for Aquatic Vegetable Brasenia schreberi. *Agronomy* **2025**, *15*, 1451. [CrossRef]
20. Feng, Q.; Zhao, C.; Li, T.; Chen, L.; Guo, X.; Xie, F.; Xiong, Z.; Chen, K.; Liu, C.; Yan, T. Design and test of a four-arm apple harvesting robot. *Trans. CSAE* **2023**, *39*, 25–33.
21. Manakitsa, N.; Maraslidis, G.S.; Moysis, L.; Fragulis, G.F. A Review of Machine Learning and Deep Learning for Object Detetion, Semantic Segmentation, and Human Action Recognition in Machine and Robotic Vision. *Technologies* **2024**, *12*, 15. [CrossRef]
22. Zhao, S.; Peng, Y.; Liu, J.; Wu, S. Tomato Leaf Disease Diagnosis Based on Improved Convolution Neural Network by Attention Module. *Agriculture* **2021**, *11*, 651. [CrossRef]
23. Peng, Y.; Wang, A.; Liu, J.; Faheem, M. A Comparative Study of Semantic Segmentation Models for Identification of Grape with Different Varieties. *Agriculture* **2021**, *11*, 997. [CrossRef]
24. Liu, J.; Abbas, I.; Noor, R.S. Development of Deep Learning-Based Variable Rate Agrochemical Spraying System for Targeted Weeds Control in Strawberry Crop. *Agronomy* **2021**, *11*, 1480. [CrossRef]
25. Peng, Y.; Zhao, S.; Liu, J. Fused-Deep-Features Based Grape Leaf Disease Diagnosis. *Agronomy* **2021**, *11*, 2234. [CrossRef]
26. Rehman, M.M.U.; Liu, J.; Nijabat, A.; Faheem, M.; Wang, W.; Zhao, S. Leveraging Convolutional Neural Networks for Disease Detection in Vegetables: A Comprehensive Review. *Agronomy* **2024**, *14*, 2231. [CrossRef]
27. Ma, J.; Li, M.; Fan, W.; Liu, J. State-of-the-Art Techniques for Fruit Maturity Detection. *Agronomy* **2024**, *14*, 2783. [CrossRef]
28. Gao, X.; Gao, J.; Qureshi, W.A. Applications, Trends, and Challenges of Precision Weed Control Technologies Based on Deep Learning and Machine Vision. *Agronomy* **2025**, *15*, 1954. [CrossRef]
29. Guo, Z.; Xiao, H.; Dai, Z.; Wang, C.; Sun, C.; Watson, N.; Povey, M.; Zou, X. Identification of apple variety using machine vision and deep learning with Multi-Head Attention mechanism and GLCM. *J. Food Meas. Charact.* **2025**, *19*, 5640–6558. [CrossRef]
30. Xiao, F.; Wang, H.; Li, Y.; Cao, Y.; Lv, X.; Xu, G. Object Detection and Recognition Techniques Based on Digital Image Processing and Traditional Machine Learning for Fruit and Vegetable Harvesting Robots: An Overview and Review. *Agronomy* **2023**, *13*, 639. [CrossRef]
31. Zhang, H.; Wang, B.; Su, L.; Yu, Z.; Liu, X.; Meng, X.; Zhao, K.; He, X. Current Research Status and Development Trends of Key Technologies for Pear Harvesting Robots. *Agronomy* **2025**, *15*, 2163. [CrossRef]
32. Charisis, C.; Argyropoulos, D. Deep learning-based instance segmentation architectures in agriculture: A review of the scopes and challenges. *Smart Agric. Technol.* **2024**, *8*, 100448. [CrossRef]
33. Khan, Z.; Shen, Y.; Liu, H. Object Detection in Agriculture: A Comprehensive Review of Methods, Applications, Challenges, and Future Directions. *Agriculture* **2025**, *15*, 1351. [CrossRef]
34. Lv, J.; Xu, H.; Xu, L.; Zou, L.; Rong, H.; Yang, B.; Niu, L.; Ma, Z. Recognition of fruits and vegetables with similar-color background in natural environment: A survey. *J. Field Robot.* **2022**, *39*, 886–902. [CrossRef]
35. Ohri, K.; Kumar, M. Review on self-supervised image recognition using deep neural networks. *Knowl. Based Syst.* **2021**, *224*, 107090. [CrossRef]
36. Tai, S.; Tang, Z.; Li, B.; Wang, S.; Guo, X. Intelligent Recognition and Automated Production of Chili Peppers: A Review Addressing Varietal Diversity and Technological Requirements. *Agriculture* **2025**, *15*, 1200. [CrossRef]
37. Kanna, K.; Ramalingam, K.; Pazhanivelan, P.; Ramasamy, J.; Prabu, P.C. YOLO deep learning algorithm for object detection in agriculture: A review. *J. Agric. Eng.* **2024**, *55*, 1641. [CrossRef]

38. Biffi, L.J.; Mitishita, E.A.; Liesenberg, V.; Centeno, J.A.S.; Schimalski, M.B.; Rufato, L. Evaluating the Performance of a Semi-Automatic Apple Fruit Detection in a High-Density Orchard System Using Low-Cost Digital RGB Imaging Sensor. *Bol. Ciênc. Geod.* **2021**, *27*, 1–20. [CrossRef]
39. Liu, W.; Sun, H.; Xia, Y.; Kang, J. Real-Time Cucumber Target Recognition in Greenhouse Environments Using Color Segmentation and Shape Matching. *Appl. Sci.* **2024**, *14*, 1884. [CrossRef]
40. Yang, L.; Mu, D.; Xu, Z.; Huang, K. Apple Surface Defect Detection Based on Gray Level Co-Occurrence Matrix and Retinex Image Enhancement. *Appl. Sci.* **2023**, *13*, 12481. [CrossRef]
41. Lei, H.; Li, C.; Tang, Y.; Zhong, Z.; Jiao, Z. Accurate and rapid image segmentation method for bayberry automatic picking via machine learning. *Int. J. Agric. Biol. Eng.* **2023**, *16*, 246–254. [CrossRef]
42. Xia, X.; Zhou, G.; Qiu, Y.; Li, Z.; Wang, J.; Hu, L.; Fan, J.; Guo, X. Detection of Young Green Apples in Orchard Environment Using Adaptive Ratio Chromatic Aberration and HOG-SVM. In *Proceedings of the 13th International Conference on Computer Vision Systems (ICVS 2017), Shenzhen, China, 10–13 July 2017*; Liu, M., Chen, H., Vincze, M., Eds.; Springer: Cham, Switzerland, 2017; pp. 253–266.
43. Xu, Y.; Imou, K.; Kaizu, Y.; Saga, K. Two-stage approach for detecting slightly overlapping strawberries using HOG descriptor. *Biosyst. Eng.* **2013**, *115*, 144–153. [CrossRef]
44. Li, Y.; Wang, H.; Zhou, W.; Xue, Z. Monocular vision and calculation of regular three-dimensional target pose based on Otsu and Haar-feature AdaBoost classifier. *Int. J. Agric. Biol. Eng.* **2020**, *13*, 171–180. [CrossRef]
45. Zhang, Y.; Liu, M.; Gong, J.; Lan, Y. Apple recognition based on two-level segmentation and region-marked gradient Hough circle transform. *Trans. CSAE* **2022**, *38*, 110–121.
46. Ho, T.-T.; Kusuma, R.M.I.; Ho, V.L.; Wen, H.Y. A Machine Learning-Based Model for Classifying the Shape of Tomato. *AgriEngineering* **2025**, *7*, 373. [CrossRef]
47. Yin, S.; Meng, L.; Liu, J. A New Apple Segmentation and Recognition Method Based on Modified Fuzzy C-means and Hough Transform. *J. Inf. Process. Syst.* **2019**, *22*, 349–354.
48. Barnea, E.; Mairon, R.; Ben-Shahar, O. Colour-agnostic shape-based 3D fruit detection for crop harvesting robots. *Biosyst. Eng.* **2016**, *146*, 57–70. [CrossRef]
49. Shakya, A.K.; Vidyarathi, A. Comprehensive Study of Compression and Texture Integration for Digital Imaging and Communications in Medicine Data Analysis. *Technologies* **2024**, *12*, 17. [CrossRef]
50. Freitas, P.G.; da Eira, L.P.; Silva, S.S.; de Farias, M.C.Q. No-Reference Image Quality Assessment Using Texture Descriptors. *J. Imaging.* **2018**, *4*, 114. [CrossRef]
51. Nanda, M.A.; Rosalinda, S.; Budiarto, R.; Novianty, I.; Salim, T.I.; Purwandoko, P.B.; Al Riza, D.F. Implementation of laser-light backscattering imaging for authentication of the geographic origin of Indonesia region citrus. *Smart Agric. Technol.* **2024**, *9*, 100527. [CrossRef]
52. Ropelewska, E. The application of image processing for cultivar discrimination of apples based on texture features of the skin, longitudinal section and cross-section. *Eur. Food Res. Technol.* **2021**, *247*, 1319–1331. [CrossRef]
53. Shen, M.; Bao, H.; Zhang, Y. Research on Hyperspectral Image Detection and Recognition of Pepper Early Blight Incubation Period Based on Spectral and Texture Features. *Prog. Biochem. Biophys.* **2025**, *52*, 233–243.
54. Ifmalinda; Andasuryani; Putri, I. Design of orange grading system based on real time image processing. *IOP Conf. Ser. Earth Environ. Sci.* **2021**, *644*, 012078. [CrossRef]
55. Zhang, X.; Toudeshki, A.; Ehsani, R.; Li, H.; Zhang, W.; Ma, R. Yield estimation of citrus fruit using rapid image processing in natural background. *Smart Agric. Technol.* **2022**, *2*, 100027. [CrossRef]
56. Yang, Q.; Chen, C.; Dai, J.; Xun, Y.; Bao, G. Tracking and recognition algorithm for a robot harvesting oscillating apples. *Int. J. Agric. Biol. Eng.* **2020**, *13*, 163–170. [CrossRef]
57. Yu, L.; Xiong, J.; Fang, X.; Yang, Z.; Chen, Y.; Lin, X.; Chen, S. A litchi fruit recognition method in a natural environment using RGB-D images. *Biosyst. Eng.* **2021**, *204*, 50–63. [CrossRef]
58. Kurtulmus, F.; Lee, W.S.; Vardar, A. Green citrus detection using 'eigenfruit', color and circular Gabor texture features under natural outdoor conditions. *Comput. Electron. Agric.* **2011**, *78*, 140–149. [CrossRef]
59. Lu, J.; Lee, W.S.; Gan, H.; Hu, X. Immature citrus fruit detection based on local binary pattern feature and hierarchical contour analysis. *Biosyst. Eng.* **2018**, *171*, 78–90. [CrossRef]
60. Fujinaga, T.; Yasukawa, S.; Ishii, K. Tomato Growth State Map for the Automation of Monitoring and Harvesting. *J. Robot. Mechatron.* **2020**, *32*, 1279–1291. [CrossRef]
61. Bai, Y.; Mao, S.; Zhou, J.; Zhang, B. Clustered tomato detection and picking point location using machine learning-aided image analysis for automatic robotic harvesting. *Precis. Agric.* **2023**, *24*, 727–743. [CrossRef]
62. Lin, G.; Tang, Y.; Zou, X.; Xiong, J.; Fang, Y. Contour-based fruit detection in natural environments using partial shape matching and probabilistic Hough transform. *Precis. Agric.* **2020**, *21*, 160–177. [CrossRef]
63. Baculo, M.J.C.; Marcos, N. Automatic Mango Detection using Image Processing and HOG-SVM. In Proceedings of the 2018 International Conference on Network, Communication and Computing (ICNCC 2018), Taipei City, Taiwan, 14–16 December 2018.

64. Peng, Y.; Zhao, S.; Liu, J. Fused Deep Features-Based Grape Varieties Identification Using Support Vector Machine. *Agriculture* **2021**, *11*, 869. [CrossRef]
65. Zhao, C.; Lee, W.S.; He, D. Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove. *Comput. Electron. Agric.* **2016**, *124*, 243–253. [CrossRef]
66. Tamilarasi, T.; Muthulakshmi, P.; Ashtiani, S.-H.M. Smart Edge Computing Framework for Real-Time Brinjal Harvest Decision Optimization. *AgriEngineering* **2025**, *7*, 196. [CrossRef]
67. Wang, J.; Sun, W. Cluster segmentation and stereo vision-based apple localization algorithm for robotic harvesting. *Front. Plant Sci.* **2025**, *16*, 1598414. [CrossRef] [PubMed]
68. Wakchaure, G.C.; Nikam, S.B.; Barge, K.R.; Kumar, S.; Meena, K.K.; Nagalkar, V.J.; Choudhari, J.D.; Kad, V.P.; Reddy, K.S. Maturity stages detection prototype device for classifying custard apple (*Annona squamosa* L.) fruit using image processing approach. *Smart Agric. Technol.* **2024**, *7*, 100394. [CrossRef]
69. Doosti-Irani, O.; Aghkhani, M.H.; Golzarian, M.R. Sweet Pepper Detection Using Fast Point Features Histogram and Unsupervised Learning. *J. Agric. Mach.* **2025**, *15*, 379–395.
70. Mohd Ali, M.; Hashim, N.; Abd Aziz, S.; Lasekan, O. Characterisation of Pineapple Cultivars under Different Storage Conditions Using Infrared Thermal Imaging Coupled with Machine Learning Algorithms. *Agriculture* **2022**, *12*, 1013. [CrossRef]
71. Ayala-Niño, D.; González-Camacho, J.M. Algoritmos de aprendizaje automático para identificar variedades de durazno con base en descriptores cromáticos y morfológicos. *Rev. Fitotec. Mex.* **2024**, *47*, 62.
72. Zhang, C.; Zou, K.; Pan, Y. A Method of Apple Image Segmentation Based on Machine Learning. *Agronomy* **2020**, *10*, 972. [CrossRef]
73. Tao, Y.; Zhou, J. Automatic apple recognition based on the fusion of color and 3D feature for robotic fruit picking. *Comput. Electron. Agric.* **2018**, *153*, 388–396. [CrossRef]
74. Kusumam, K.; Krajnik, T.; Pearson, S.; Duckett, T.; Cielniak, G. 3D-vision based detection, localization, and sizing of broccoli heads in the field. *J. Field Robot.* **2017**, *34*, 1505–1518. [CrossRef]
75. Han, X.; Ye, H.; Zhang, Y.; Nie, C.; Wen, F. Study on the Method of Vineyard Extraction from GF-6 Remote Sensing Imagery Based on the Boruta Algorithm and Machine Learning. *Remote Sens.* **2024**, *14*, 2542.
76. Lamptey, F.P.; Teye, E.; Abano, E.E.; Amuah, C.L.Y. Application of handheld NIR spectrometer for simultaneous identification and quantification of quality parameters in intact mango fruits. *Smart Agric. Technol.* **2023**, *6*, 100357. [CrossRef]
77. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
78. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.1.
79. Jocher, G. Ultralytics YOLOv5. 2020. Available online: https://github.com/ultralytics/yolov5 (accessed on 21 July 2026).
80. Yaseen, M. What is YOLOv8: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector. *arXiv* **2024**, arXiv:2408.15857.
81. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. In *Computer Vision—ECCV 2024*; Springer: Cham, Switzerland, 2024; pp. 1–21.
82. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*; Curran Associates: Red Hook, NY, USA, 2024; pp. 107984–108011.
83. Khanam, R.; Hussain, M. YOLOv11: An Overview of the Key Architectural Enhancements. *arXiv* **2024**, arXiv:2410.17725.
84. Jocher, G.; Qiu, J. Ultralytics YOLOv26: Unified Real-Time End-to-End Vision Models. *arXiv* **2026**, arXiv:2606.03748.
85. Shi, X.; Wang, S.; Zhang, B.; Ding, X.; Qi, P.; Qu, H.; Li, N.; Wu, J.; Yang, H. Advances in Object Detection and Localization Techniques for Fruit Harvesting Robots. *Agronomy* **2025**, *15*, 145. [CrossRef]
86. Ma, J.; Feng, W.; Zhou, Y.; Liu, J. An Improved YOLOv8 Model for Lotus Seedpod Instance Segmentation in the Lotus Pond Environment. *Comput. Electron. Agric.* **2024**, *225*, 109090.
87. Ji, W.; Wang, J.; Xu, B.; Zhang, T. Apple Grading Based on Multi-Dimensional View Processing and Deep Learning. *Foods* **2023**, *12*, 2117. [CrossRef] [PubMed]
88. Jiang, L.; Wang, Y.; Wu, C.; Wu, H. Fruit Distribution Density Estimation in YOLO-Detected Strawberry Images: A Kernel Density and Nearest Neighbor Analysis Approach. *Agriculture* **2024**, *14*, 1848. [CrossRef]
89. Sun, X.; Wang, L.; Zheng, Y.; Zhu, H.; Sui, Y.; Ma, Z.; Guo, R.; Hu, W.; Zhang, T.; Yan, P.; et al. Accurate apple fruit stalk cutting technology based on improved YOLOv8 with dual cameras. *Appl. Eng. Agric.* **2025**, *41*, 97–107. [CrossRef]
90. Sun, C.; Zhang, L.; Zhai, L.; Shen, T.; Cai, J.; Zou, X.; Guo, Z. Automatic early bruise detection in strawberry fruit by hyperspectral imaging and deep learning techniques. *Postharvest Biol. Technol.* **2026**, *232*, 113966. [CrossRef]
91. Xu, B.; Cui, X.; Ji, W.; Yuan, H.; Wang, J. Apple Grading Method Design and Implementation for Automatic Grader Based on Improved YOLOv5. *Agriculture* **2023**, *13*, 124. [CrossRef]

92. Zhao, S.; Liu, J.; Hua, T.; Jiang, Y. Improved UNet Recognition Model for Multiple Strawberry Pests Based on Small Samples. *Agronomy* **2025**, *15*, 2252. [CrossRef]
93. Zhou, X.; Chen, W.; Wei, X. Improved Field Obstacle Detection Algorithm Based on YOLOv8. *Agriculture* **2024**, *14*, 2263. [CrossRef]
94. Zhu, H.; Wang, D.; Wei, Y.; Zhang, X.; Li, L. Combining Transfer Learning and Ensemble Algorithms for Improved Citrus Leaf Disease Classification. *Agriculture* **2024**, *14*, 1549. [CrossRef]
95. Ma, Z.; Zhang, N.; Wang, S.; Li, Y.; Pan, Y.; Zhang, J.; Liu, C.; Gao, H. A lightweight real-time potato damage detection method based on improved YOLOv8. *J. Food Meas. Charact.* **2025**, *19*, 5931-5945. [CrossRef]
96. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
97. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
98. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976.
99. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for realtime object detectors. In Proceedings of the IEEE/CVF Conference on Computer Visionand Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
100. Shobaki, W.A.; Milanova, M. A Comparative Study of YOLO, SSD, Faster R-CNN, and More for Optimized Eye-Gaze Writing. *Sci* **2025**, *7*, 47. [CrossRef]
101. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conf on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
102. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
103. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the 29th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015.
104. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
105. Xiao, F.; Wang, H.; Xu, Y.; Zhang, R. Fruit Detection and Recognition Based on Deep Learning for Automatic Harvesting: An Overview and Review. *Agronomy* **2023**, *13*, 1625. [CrossRef]
106. Tanimoto, Y.; Zhang, Z.; Yoshida, S. Object Detection for Yellow Maturing Citrus Fruits from Constrained or Biased UAV Images: Performance Comparison of Various Versions of YOLO Models. *AgriEngineering* **2024**, *6*, 4308–4324. [CrossRef]
107. Divyanth, L.G.; Rathore, D.; Senthilkumar, P.; Patidar, P.; Zhang, X.; Karkee, M.; Machavaram, R.; Soni, P. Estimating depth from RGB images using deep-learning for robotic applications in apple orchards. *Smart Agric. Technol.* **2023**, *6*, 100345. [CrossRef]
108. Wang, Z.; Cui, W.; Huang, C.; Zhou, Y.; Zhao, Z.; Yue, Y.; Dong, X.; Lv, C. Framework for Apple Phenotype Feature Extraction Using Instance Segmentation and Edge Attention Mechanism. *Agriculture* **2025**, *15*, 305. [CrossRef]
109. Wu, F.; Zhang, R.; Mo, F.; Qin, J.; Yang, X.; Li, J.; Zou, X. An Improved CycleGAN Approach for Nighttime Pineapple Detection with Limited Datasets. *Agriculture* **2024**, *14*, 2200.
110. Cang, H.; Yan, T.; Duan, L.; Yan, J.; Zhang, Y.; Tan, F.; Lv, X.; Gao, P. Jujube quality grading using a generative adversarial network with an imbalanced data set. *Biosyst. Eng.* **2023**, *236*, 224–237. [CrossRef]
111. Sapkota, R.; Meng, Z.; Karkee, M. Synthetic meets authentic: Leveraging LLM generated datasets for YOLO11 and YOLOv10-based apple detection through machine vision sensors. *Smart Agric. Technol.* **2024**, *9*, 100614. [CrossRef]
112. Mullins, C.C.; Esau, T.J.; Zaman, Q.U.; Hennessy, P.J. Optimizing data collection requirements for machine learning models in wild blueberry automation through the application of DALL-E 2. *Smart Agric. Technol.* **2025**, *10*, 100764. [CrossRef]
113. Ushiroji, T.; Fujinaga, T. Automatic generation of synthetic data and object detection datasets in virtual environments based on tomato-harvesting robot vision. *Smart Agric. Technol.* **2026**, *14*, 101947. [CrossRef]
114. Zhou, L.; Jin, S.; Wang, J.; Zhang, H.; Shi, M.; Zhou, H. 3D positioning of Camellia oleifera fruit-grabbing points for robotic harvesting. *Biosyst. Eng.* **2024**, *246*, 110–121. [CrossRef]
115. Chen, X.; Shi, Y.; Li, X. Research on recognition method of Chinese cabbage growth periods based on Swin Transformer and transfer learning. *Appl. Eng. Agric.* **2023**, *39*, 381–390. [CrossRef]
116. Zhao, J.; Lipani, A.; Schillaci, C. Fallen apple detection as an auxiliary task: Boosting robotic apple detection performance through multi-task learning. *Smart Agric. Technol.* **2024**, *8*, 100436. [CrossRef]
117. Wang, L.; Zhao, Y.; Liu, S.; Li, Y.; Chen, S.; Lan, Y. Precision Detection of Dense Plums in Orchards Using the Improved YOLOv4 Model. *Front. Plant Sci.* **2022**, *13*, 839269. [CrossRef] [PubMed]
118. Ufuah, D.; Thomas, G.; Balocco, S.; Manickavasagan, A. Data Augmentation Based on Generative Adversarial Networks for Date Fruit Classification. *Appl. Eng. Agric.* **2022**, *38*, 975–982. [CrossRef]

119. Espejo-Garcia, B.; Mylonas, N.; Athanasakos, L.; Vali, E.; Fountas, S. Combining generative adversarial networks and agricultural transfer learning for weeds identification. *Biosyst. Eng.* **2021**, *204*, 79–89. [CrossRef]
120. Wang, A.; Xu, Y.; Zhang, L.; Zhu, Q.; Li, A.; Liu, J. Tomato Yield Estimation Using an Improved Lightweight YOLO11n Network and an Optimized Region TrackingCounting Method. *Agriculture* **2025**, *15*, 1353. [CrossRef]
121. Zheng, S.; Lin, Y.; Weng, W.; Jia, X.; Yu, S.; Wu, Z. Tomato Recognition and Localization Method Based on Improved YOLOv5n-seg Model and Binocular Stereo Vision. *Agriculture* **2023**, *13*, 2339. [CrossRef]
122. Shi, Y.; Qing, S.; Zhao, L.; Wang, F.; Yuwen, X.; Qu, M.; Duan, Z. YOLOv9s-Pear: A Lightweight Model for Young Red Pear Recognition Based on Improved YOLOv9s. *Agronomy* **2024**, *14*, 1625–1640. [CrossRef]
123. Guan, X.; Shi, L.; Yang, W.; Ge, H.; Wei, X.; Ding, Y. Multi-Feature Fusion Recognition and Localization Method for Unmanned Harvesting of Aquatic Vegetables. *Agriculture* **2024**, *14*, 971. [CrossRef]
124. Tao, S.; Wen, S.; Hu, G.; Ge, Y.; Wen, J.; Cao, X.; Chen, J. Improved YOLOv8n-based detection of grapes in orchards. *INMATEH-Agric. Eng.* **2024**, *74*, 529–540.
125. Zhang, T.; Zhou, J.; Liu, W.; Yue, R.; Yao, M.; Shi, J.; Hu, J. Seedling-YOLO: High-Efficiency Target Detection Algorithm for Field Broccoli Seedling Transplanting Quality Based on YOLOv7-Tiny. *Agronomy* **2024**, *14*, 931. [CrossRef]
126. Zhao, Y.; Zhang, X.; Sun, J.; Yu, T.; Cai, Z.; Zhang, Z.; Mao, H. Low-Cost Lettuce Height Measurement Based on Depth Vision and Lightweight Instance Segmentation Model. *Agriculture* **2024**, *14*, 1596. [CrossRef]
127. Zuo, Z.; Gao, S.; Peng, H.; Xue, Y.; Han, L.; Ma, G.; Mao, H. Lightweight Detection of Broccoli Heads in Complex Field Environments Based on LDBC-YOLO. *Agriculture* **2024**, *14*, 2359. [CrossRef]
128. Shi, Q.; Zhang, Y.; Du, X.; Chen, T.; Wang, Y. Light-YOLO-Pepper: A Lightweight Model for Detecting Missing Seedlings. *Agriculture* **2024**, *14*, 1957–1973.
129. Leng, T.; Zhang, Y.; Fan, Y.; Gao, H.; Zhou, B.; Mu, J. A Lightweight Shape-Aware YOLO Network for Field Strawberry Maturity Detection Under Complex Orchard Conditions. *Agronomy* **2026**, *16*, 1074. [CrossRef]
130. Zhao, W.; Ding, X.; Wang, J. SDGD-YOLO: A lightweight detection model for assessing the ripeness of black raspberries in complex greenhouse environments. *Front. Plant Sci.* **2026**, *17*, 1843631. [CrossRef] [PubMed]
131. Lingamgunta, S.; Muddana, J.; Vincent, D.R.; S.A.M., F. Distilled vision transformers with CNN fusion for robust cashew apple maturity prediction. *Front. Plant Sci.* **2026**, *17*, 1787609. [CrossRef] [PubMed]
132. Li, A.; Wang, C.; Ji, T.; Wang, Q.; Zhang, T. D^3-YOLOv10: Improved YOLOv10-Based Lightweight Tomato Detection Algorithm Under Facility Scenario. *Agriculture* **2024**, *14*, 2268. [CrossRef]
133. Wang, D.; He, D.J. Apple detection and instance segmentation in natural environments using an improved Mask Scoring R-CNN Model. *Front. Plant Sci.* **2022**, *13*, 1016470. [CrossRef] [PubMed]
134. Jiang, H.K.; Liu, J.Z.; Lei, X.J.; Xu, B.C.; Jin, Y.C. Multi-stage fusion of dual attention mask R-CNN and geometric filtering for fast and accurate localization of occluded apples. *Artif. Intell. Agric.* **2026**, *16*, 187–205. [CrossRef]
135. Ji, W.; Zhang, T.; Xu, B.; He, G. Apple recognition and picking sequence planning for harvesting robot in a complex environment. *J. Agric. Eng.* **2024**, *55*, 1549. [CrossRef]
136. Ji, W.; Gao, X.X.; Xu, B.; Pan, Y.; Zhang, Z.; Zhao, D.A. Apple target recognition method in complex environment based on improved YOLOv4. *J. Food Process Eng.* **2021**, *44*, e13866. [CrossRef]
137. Li, A.; Wang, C.; Wang, A.C.; Sun, J.P.; Gu, F.W.; Zhang, T. YOLO-MSRF: A Multimodal Segmentation and Refinement Framework for Tomato Fruit Detection and Segmentation with Count and Size Estimation Under Complex Illumination. *Agriculture* **2026**, *16*, 277. [CrossRef]
138. Sun, J.; He, X.F.; Ge, X.; Wu, X.H.; Shen, J.F.; Song, Y.Y. Detection of key organs in tomato based on deep migration learning in a complex background. *Agriculture* **2018**, *8*, 196. [CrossRef]
139. Zhao, S.Y.; Fang, C.; Hua, T.; Jiang, Y. Detecting the maturity of red strawberries using improved YOLOv8s model. *Agriculture* **2025**, *15*, 2263. [CrossRef]
140. Xie, H.H.; Zhang, Z.Y.; Zhang, K.L.; Yang, L.; Zhang, D.X.; Yu, Y. Research on the visual location method for strawberry picking points under complex conditions based on composite models. *J. Sci. Food Agric.* **2024**, *104*, 8566–8579. [CrossRef] [PubMed]
141. Shen, L.; Su, J.Y.; Huang, R.; Quan, W.M.; Song, Y.Y.; Fang, Y.L.; Su, B.F. Fusing attention mechanism with Mask R-CNN for instance segmentation of grape cluster in the field. *Front. Plant Sci.* **2022**, *13*, 934450. [CrossRef] [PubMed]
142. Hou, C.J.; Xu, J.L.; Tang, Y.; Zhuang, J.J.; Tan, Z.P.; Chen, W.L.; Wei, S.; Huang, H.S.; Fang, M.W. Detection and localization of citrus picking points based on binocular vision. *Precis. Agric.* **2024**, *25*, 2321–2355. [CrossRef]
143. Liu, Y.Y.; Ren, H.M.; Zhang, Z.; Men, F.S.; Zhang, P.Y.; Wu, D.L.; Feng, R.Z. Research on multi-cluster green persimmon detection method based on improved Faster RCNN. *Front. Plant Sci.* **2023**, *14*, 117114. [CrossRef] [PubMed]

https://doi.org/10.3390/agriculture16171924